

Aalto-yliopisto
Sähkötekniikan korkeakoulu
Tietoliikennetekniikan koulutusohjelma

Jaakko Möller

Monilähetysjärjestelmä erityisvälitysverkoille

Diplomityö
Espoo, 25. elokuuta 2014

Valvoja: Professori Jukka Manner
Ohjaaja: Diplomi-insinööri Risto Järvinen

Aalto University
 School of Electrical Engineering
 Degree Programme in Communications Engineering

ABSTRACT OF
 MASTER'S THESIS

Author:	Jaakko Möller		
Title:	Multicast System for Challenged Networks		
Date:	August 25, 2014	Pages:	viii + 70
Major:	Networking Technology	Code:	S-38
Supervisor:	Professor Jukka Manner		
Advisor:	Risto Järvinen M.Sc. (Tech.)		
<p>In this thesis we present a multicast system for challenged networks. Challenged networks include networks used in crisis operation areas as well as networks intended for official government use. A typical challenged network is composed of an especially weakly connected edge network, while offering a better connectivity in the core network. Additionally, the nodes of the network are typically in continuous movement, especially in the edge of the network.</p> <p>The multicast system was implemented as part of MICS (Multi Interface Communications Software), which is a messaging system designed for challenged networks. We study the MICS-system’s architecture in order to defer the best ways to implement the multicast system into MICS, taking into account the existing components.</p> <p>A special test system was implemented to test the implementation. The test results show that the multicast system is able to offer a functioning multicast service as part of the MICS-system. The multicast system should, however, be developed further as MICS-system evolves.</p>			
Keywords:	multicast routing, challenged networks, MICS, delay tolerant networks, MANET		
Language:	Finnish		

Aalto-yliopisto
 Sähkötekniikan korkeakoulu
 Tietoliikennetekniikan koulutusohjelma

DIPLOMITYÖN
 TIIVISTELMÄ

Tekijä:	Jaakko Möller		
Työn nimi:	Monilähetysjärjestelmä erityisvälitysverkoille		
Päiväys:	25. elokuuta 2014	Sivumäärä:	viii + 70
Pääaine:	Tietoverkkotekniikka	Koodi:	S-38
Valvoja:	Professori Jukka Manner		
Ohjaaja:	Diplomi-insinööri Risto Järvinen		
<p>Tässä työssä esitetään monilähetysjärjestelmä erityisvälitysverkoille. Erityisvälitysverkoilla tarkoitetaan erityisesti kriisitilanteisiin, sekä julkishallintoon tarkoitettuja verkkoja, joissa tiedonsiirtoyhteydet ovat tyypillisesti erityisen haastavat verkon laidalla, verkon keskiosan tarjotessa paremmin yhdistetyn runkoverkon. Verkon solmut ovat myös tyypillisesti jatkuvassa liikkeessä, erityisesti verkon laidalla.</p> <p>Monilähetysjärjestelmä toteutettiin osaksi viivesietoisille verkoille toteutettua MICS-viestivälitysjärjestelmää (eng. Multi Interface Communications Software). Työssä käydään läpi MICS-järjestelmän arkkitehtuuria, sekä arvioidaan miten monilähetyspalvelu on järkevintä toteuttaa MICS-järjestelmään, ottaen huomioon järjestelmän olemassa olevat komponentit.</p> <p>Toteutus testattiin työtä varten toteutetulla testausjärjestelmällä. Testien tuloksena voidaan todeta, että toteutettu monilähetysjärjestelmä kykenee toteuttamaan toimivan monilähetyspalvelun MICS-järjestelmään. Monilähetyspalvelua kannattaa kuitenkin kehittää edelleen MICS-järjestelmän keskeisten komponenttien kehittyessä pidemmälle.</p>			
Asiasanat:	monilähetysreititys, haasteelliset verkot, MICS, viivesietoiset verkot, MANET		
Kieli:	Suomi		

Kiitokset

Haluaisin kiittää ohjaajaani ja valvojaani hyvästä ohjauksesta ja arvokkaasta palautteesta.

Helsinki, 25. elokuuta 2014

Jaakko Möller

Lyhenteet

BGP	Border Gateway Protocol. IP-verkkojen autonomisten järjestelmien (eng. Autonomous System) välillä käytetty reititysprotokolla.
CF	Classical Flooding. Tulvitusmekanismi, jossa kaikki vastaanottavat solmut tulvittavat paketin.
D-Bus	Prosessien väliseen kommunikointiin tarkoitettu järjestelmä.
DTN	Delay Tolerant Network. Viivesietoinen verkko.
E-CDS	Essential Connected Dominating Set. Yksi SMF-protokollan algoritmivaihtoehtoista MCDS:n selvittämiseen.
GPS	Global Positioning System. Yhdysvaltain Puolustusministeriön kehittämä ja ylläpitämä satelliittipaikannusjärjestelmä.
GSM	Global System for Mobile Communications. Maailmanlaajuisesti yleisesti käytetty matkapuhelinjärjestelmä.
HF	High Frequency. Radiotaajuuudet 3-30 MHz.
IP	Internet Protocol. Internet-protokolla, pakettivälitysprotokolla erilaisten verkkojen yhdistämiseen.
IS-IS	Intermediate System to Intermediate System. Internet standardissa RFC 1142 esitetty reititysprotokolla verkkojen sisäiseen reititykseen.
LBM	Location-Based Multicast. Geolähetysprotokolla, jossa paketit edelleenvälitetään joko edelleenvälitysalueiden tai puhtaasti solmujen sijaintien perusteella.
MANET	Mobile Adhoc Network. Tiettyä tarkoitusta varten luotu langaton verkko, jonka jokainen jäsen on reitittävä.

MCDS	Minimum Connected Dominating Set. Minimaalinen verkon yhdistävä joukko.
MID	Message Identifier. Viestitunniste. Viestin kenttä, jonka tarkoituksena on yksilöidä viesti muista viesteistä.
MPR	Multipoint Relay. Tulvittava solmu.
MPR-CDS	Multipoint Relay Connected Dominating Set. Yksi SMF-protokollan algoritmivaihtoehdoista MCDS:n selvittämiseen.
MRIB	Multicast Routing Information Base. Monilähetysreititystietokanta.
NHDP	Neighborhood Discovery Protocol. Naapuruston selvittämiseen tarkoitettu protokolla.
Netlink	Internet standardissa RFC 3549 kuvattu protokolla käyttöjärjestelmän ja sovellusten väliseen, sekä sovellusten keskinäiseen kommunikointiin.
OLSR	Optimized Link State Protocol. MANET verkkoihin suunniteltu täsmäreititysprotokolla, joka on esitetty Internet-standardissa RFC 3626.
OSPF	Open Shortest Path First. Yksi IP-verkkojen autonomisten järjestelmien sisällä käytetyistä reititysprotokollista.
PIM	Protocol Independent Multicast. Protokollariippumaton monilähetys.
PIM-DM	Protocol Independent Multicast - Sparse Mode. Protokollariippumaton monilähetys - harva tila. Internet-standardissa RFC 4601 esitetty monilähetysprotokolla.
PIM-SM	Protocol Independent Multicast. Protokollariippumaton monilähetys - tiheä tila. Internet-standardissa RFC 3973 esitetty monilähetysprotokolla.
RFC	Request for Comments. Internetin standardointi elimen IETF:n standardi.
RIB	Routing Information Base. Reititystietokanta.
RIP	Routing Information Protocol. Yksi IP-verkkojen autonomisten järjestelmien sisällä käytetty reititysprotokolla.
RIPv2	Routing Information Protocol version 2. Yksi IP-verkkojen autonomisten järjestelmien sisällä käytetty reititysprotokolla. Kehittyneempi versio RIP-protokollasta.

S-MPR	Source-Based Multipoint Relay. Yksi SMF-protokollan algoritmivaihtoehtoista MCDS:n selvittämiseen.
SMF	Simplified Multicast Forwarding. Internet-standardissa RFC 6621 esitelty monilähetysreititysprotokolla MANET verkkoihin.
SMS	Short Message Service. Matkapuhelinten tekstiviestijärjestelmä.
TETRA	Terrestrial Trunked Radio. Viranomaiskäyttöön tarkoitettu digitaalinen radioverkko.
TMR	Tactical Message Replicator. Monilähetysohjelmistokomponentti, joka kykenee monilähetyksien tehokkaaseen suorittamiseen erityisvälitysverkoissa.
TTL	Time-To-Live. Paketin elinaika verkossa.
VHF	Very High Frequency. Radiotaajuuudet 30-300 MHz.
VPN	Virtual Private Network. Tekniikka, jolla lähiverkkoa voidaan laajentaa julkisen verkon yli.
WAN	Wide Area Network. Laajaverkko, verkko joka kattaa laajoja maantieteellisiä alueita.
ZRP	Zone Routing Protocol. Hierarkinen reititysprotokolla langattomiin verkkoihin.

Sisältö

Lyhenteet	v
1 Johdanto	1
2 Reititys- ja monilähetysprotokollat	6
2.1 Monilähetys IP-verkoissa	6
2.2 Reititys MANET-verkoissa	11
2.2.1 Täsmälähetys	12
2.2.2 Monilähetys	17
2.3 Reititys ja tiedonvälitys viivesietoisissa verkoissa	23
2.4 Yhteenveto	26
3 MICS-järjestelmä	28
3.1 MICS-järjestelmän arkkitehtuuri	29
3.2 Reititys	31
3.3 Monilähetys	32
3.4 Toteutus - Ydinjärjestelmän komponenttina vai sovelluksena .	36
3.5 Yhteenveto	41
4 Toteutus ja validointi	42
4.1 TMR-verkon toiminta	42
4.2 Toiminnan testaus ja arviointi	47
5 Johtopäätökset	65

Luku 1

Johdanto

Internet-verkoissa viestinvälitys suoritetaan tyypillisesti täsmälähetyksenä. Täsmälähetyksissä viestillä on yksi lähettäjä ja vastaanottaja. Viestiä välitettäessä reitittävän solmun tulee tietää mille solmulle viesti tulee välittää edelleen, jotta reititys kohdesolmulle onnistuu. Täsmälähetyksessä usealle vastaanottajalle tarkoitettu viesti tulee välittää jokaiselle vastaanottajalle erikseen. Verkossa siis välitetään samasta viestistä useampi kopio jokaista kohdesolmua kohden. Tällöin kopio lähetettävästä viestistä saatetaan joutua välittämään useamman kerran verkon linkkien yli, mikä tuhlaa verkon resursseja. Viestit eroavat ainoastaan kohteidensa osalta, joten jos viestin kuorma välitetään useamman kerran verkon linkin yli, välitetään turhaan jo aiemmin välitettyä tietoa.

Yksi vaihtoehto täsmälähetykselle on yleislähetys. Yleislähetyksessä lähetettävästä viestistä välitetään kopio kaikille verkon vastaanottajille lyhyintä reittiä hyväksikäyttäen. Tyypillisesti viestin kohteeksi asetetaan yleislähetysosoite, jolloin verkon reitittäville solmuille jää tehtäväksi reitittää viesti lyhyintä reittiä pitkin siten, että viestistä kyetään välittämään kopio kaikille verkon solmuille. Yleislähetystä käytettäessä samasta viestistä ei välitetä tuhlaavasti useampia kopioita, vaan riittää, että viestistä välitetään linkin yli yksi kopio, jonka kohteena on verkon yleislähetysosoite. Yleislähetystä käytettäessä viesti saatetaan kuitenkin välittää myös solmuille, joille viestiä ei ollut tarkoitus lähettää. Jos vain osa verkon solmuista on viestin kohteena, saatetaan yleislähetyksessä tuhlaa merkittävästi verkon resursseja.

Täsmälähetystä ja yleislähetystä tarkoituksenmukaisempi keino viestinvälitykseen useille kohdesolmuille on monilähetys. Monilähetyksessä viestistä välitetään kopio jokaiselle viestin vastaanottajalle lyhyintä reittiä hyväksikäyttäen. Tyypillisessä monilähetyspalvelumallissa verkossa varataan tietty osoi-

tealue monilähetysryhmien osoitealueeksi. Monilähetysryhmästä kiinnostunut solmu voi tilata itselleen monilähetysryhmän lähettämällä tilauspyynnön paikalliselle monilähetysreitittimelleen. Lähettävä solmu lähettää monilähetysryhmälle tarkoitetun viestin monilähetysryhmän osoitteeseen. Paikallinen monilähetysreititin vastaanottaa viestin ja välittää sen muille verkon monilähetysreitittimille siten, että kaikki ryhmän vastaanottajat kyetään saavutamaan.

Monilähetyistä käytettäessä viestejä ei siis välitetä turhaan useita kertoja verkon linkkien yli, joten se käyttää verkon resursseja tehokkaammin kuin täsmälähetys. Viestistä ei myöskään välitetä kopiota solmuille, jotka eivät ole kohteita kyseiselle monilähetysryhmälle, joten viesti välitetään tehokkaammin kuin yleislähetystä käytettäessä. Haittapuolena on lisääntynyt signaointi verkossa: verkon monilähetysreitittimien tulee signaloida toisilleen monilähetysryhmien muutokset. Lisääntyneestä signaoinnista huolimatta monilähetyistä hyväksikäyttämällä saatetaan säästää huomattavasti verkon resursseja, mikä on erityisen tärkeää jos verkon resurssit ovat vähäiset.

Erityisvälitysverkoilla tarkoitetaan haasteellisia verkkoja joissa Internet-verkkojen reititys ei enää kykene toimimaan. Erityisvälitysverkkoihin kuuluvat valtion viranomaisten, kuten poliisien, sairaanhoitajien ja palomiesten työtehtäviin tarvittavat verkot, sekä erityisiin kriisitilanteisiin tarkoitetut verkot. Erityisvälitysverkoille on ominaista rajoitettu siirtokapasiteetti ja mahdollisesti katkonainen yhteys. Erityisesti verkon laidalla käytettävien kannettavien päätelaitteiden tarjoamat yhteydet ovat huonot, kun taas verkon rungossa on tyypillisesti tarjolla paremmat tiedonsiirtoyhteydet. Erityisvälitysverkoissa välitettävät viestit ovat usein luonteeltaan erityisen tärkeitä, joten viestien perille saattaminen on tärkeää.

Erityisvälitysverkoissa verkon reitittävien solmujen tulee kantaa Internet-verkkoihin verrattuna enemmän vastuuta viestien perille saattamisesta. Internet-verkoissa lähettäjän ja vastaanottajan väliset reitittävät solmut ovat yleensä tilattomia, jolloin tila säilytetään vain lähettäjällä ja kohteella. Tällöin esimerkiksi luotettavan tiedonsiirron aikaansaaminen vaatii erillistä signaointia verkon yli, mikä on toimiva ratkaisu Internet-verkoissa, joissa siirtokapasiteettia on enemmän ja tiedonsiirron viiveet pieniä. Erityisvälitysverkoissa verkon siirtokapasiteetti on vähäinen ja yhteys katkonainen, joten lähettäjän ja kohteen välinen signaointi on kallista tai jopa mahdotonta. Tämän vuoksi erityisvälitysverkoissa reitittävien solmujen tulee varmistaa viestien perille saattaminen.

MANET-verkoilla (eng. Mobile Ad hoc Networks) tarkoitetaan tiettyyn käyttötarkoitukseen (ad hoc) tehtyjä verkkoja, joissa kommunikointi tapahtuu

tyypillisesti langattomasti. MANET-verkoille on myös tyypillistä solmujen jatkuva liike, mikä aiheuttaa jatkuvia muutoksia verkon topologiassa. MANET-verkkoja on tutkittu paljon erilaisten langattomaan viestintään kykenevien laitteiden, kuten kännyköiden ja kannettavien tietokoneiden, yleistyttyä. MANET-verkoille onkin suunniteltu useita monilähetysjärjestelmiä, jotka kykenevät selviytymään MANET-verkoille tyypillisistä jatkuvista topologiamuutoksista. Lisäksi MANET-verkkojen monilähetysprotokollat hyödyntävät verkon langattoman siirtotien ominaisuuksia.

Viivesietoisissa verkoissa (eng. Delay Tolerant Network, DTN) verkko oletetaan katkonaiseksi, siirtonopeudet pieneksi ja linkkien viiveet suuriksi. Verkon solmujen välille ei oleteta päästä päähän yhteyttä, vaan lähtökohtaisesti minkään solmun välillä ei ole suoraan reititettävää yhteyttä. Viestinvälityksessä yritetään tyypillisesti hyödyntää verkon solmujen liikettä. Liikettä hyödynnetään esimerkiksi muodostamalla kopioita lähetettävästä viestistä naapurisolmuille. Solmujen liikkuesssa verkossa, joku kopion saaneista solmuista saattaa kohdata kohdesolmun ja välittää viestin perille. Mitä enemmän kopioita muodostetaan, sitä todennäköisemmin viesti kyetään toimittamaan perille.

Erityisvälitysverkkojen kannalta Internet-, MANET-, ja DTN-verkkojen monilähetysratkaisut ovat puutteellisia. Internet-verkkojen monilähetysratkaisut eivät ole sopivia erityisvälitysverkkoihin, koska Internet-verkoissa oletetaan päästä-päähän yhteys. Tämän oletuksen vuoksi Internet-verkkojen monilähetysprotokollat eivät kykene muodostamaan monilähetysreititystä katkonaisessa verkossa. Koska Internet-maailmassa monilähetystä tarvitsevat lähinnä paljon siirtokaistaa tarvitsevat palvelut, joilla ei ole tarvetta tiedonsiirron luotettavuudelle, kuten konferenssipalvelut ja IPTV, ei luotettavuuteen ole erityisesti panostettu. Erityisvälitysverkkojen viestien tärkeyden vuoksi luotettavuuden puute on ongelmallista.

MANET-verkkojen monilähetysjärjestelmät kykenevät tarjoamaan monilähetyksen vaikka verkon topologia muuttuisi jatkuvasti. Lisäksi MANET-monilähetys hyödyntää langattoman siirtotien ominaisuuksia, jolloin yhdellä lähetyksellä viesti voidaan välittää usealle vastaanottajalle. Erityisvälitysverkkojen kannalta MANET-verkkojen monilähetysratkaisut kärsivät kuitenkin samoista ongelmista kuin IP-verkkojen monilähetysprotokollat: viestejä ei oleteta tärkeäksi ja solmujen välille oletetaan päästä-päähän yhteys.

DTN-verkot kykenevät reitittämään paketteja vaikka verkko olisi katkonainen ja solmujen välinen päästä-päähän yhteys puuttuisi. Paketinvälityksessä hyödynnetään solmujen liikettä ja paketeista otetut kopiot parantavat toimitusvarmuutta. DTN-verkoissa solmujen väliset yhteydet oletetaan huonoiksi

kaikkien verkon solmujen kesken. Erityisvälitysverkoissa yhteys on kuitenkin usein huono vain verkon laidoilla. Verkon laidalla käytetään kannettavia, langattomasti toimivia, päätelaitteita. Erityisvälitysverkoilla on usein käytettävissään paremman yhteyden tarjoava runkoverkko. DTN-verkoille suunnitellut ratkaisut olettavat kaikkien verkon solmujen yhteydet huonoiksi, eivätkä ne hyödynnä runkoverkon tarjoamaa suurempaa siirtokapasiteettia ja parempaa toimitusvarmuutta.

Koska olemassa olevat monilähetysratkaisut perustuvat joko Internet-, MANET-, tai DTN-verkkojen oletuksiin, tarvitaan erityinen monilähetysjärjestelmä erityisvälitysverkoille. Järjestelmän tulee kyetä toimittamaan monilähetyspaketit perille myös verkoissa joissa tapahtuu yhteyskatkoksia. Lisäksi järjestelmän tulee kyetä toimimaan verkoissa joissa yhteyksien siirtokapasiteetti on pieni ja sen tulee varoa ruuhkauttamasta verkkoa. Koska erityisvälitysverkoissa on usein käytössä paremman yhteyden tarjoava runkoverkko, tulee järjestelmän kyetä hyödyntämään runkoverkon ominaisuuksia.

Tässä työssä esitetään monilähetysjärjestelmä nimeltä TMR (eng. Tactical Message Replicator). TMR on monilähetysjärjestelmä erityisvälitysverkoille, joka kykenee tehokkaaseen monilähetykseen haastavissa verkko-olosuhteissa. TMR-järjestelmä sallii monilähetyksryhmien prioriteetin säätämisen siten, että järjestelmä käyttää enemmän resursseja tärkeiden viestien toimittamiseen, muttei kuitenkaan ruuhkauta verkkoa vähemmän tärkeitä viestejä välitettäessä. Järjestelmän rajapinnaksi valittiin tuttu sähköpostilistojen hallinta-rajapinta. Erityisvälitysverkon laidan pienemmän siirtokapasiteetin vuoksi TMR hyödyntää runkoverkon suurempaa siirtokapasiteettia viestien välityksessä.

TMR-monilähetysjärjestelmä toteutettiin monilähetyspalvelun tarjoavaksi komponentiksi erityisvälitysverkoille tarkoitettuun MICS-järjestelmään (Multi Interface Communications Software). TMR-monilähetysjärjestelmä toteuttaa viestinvälityksen MICS-järjestelmän tarjoamia sähköpostijärjestelmästä tuttuja IMAP- ja SMTP-rajapintoja hyväksikäyttäen. Työhön sisältyy myös testit, joiden avulla voidaan todeta toteutetun kokonaisuuden oikea toiminta.

Työn kontribuutiona on erityisvälitysverkkoihin soveltuvan monilähetysjärjestelmän suunnittelun, toteutuksen ja arvioinnin yhteydessä havaitut seikat erityisvälitysverkoissa toteutettuun monilähetykseen liittyen. Työn tuloksena voidaan todeta, että vain osa tieteessä esitetyistä monilähetyksen mekanismeista kykenee toimimaan erityisvälitysverkoissa. Lisäksi voidaan antaa ehdotuksia monilähetysjärjestelmän kehittämisestä MICS-järjestelmän kehityksessä edelleen.

Työ on järjestetty siten, että toisessa luvussa esitetään IP-, MANET- ja

DTN-verkkojen reititys- ja monilähetysreititysratkaisuja, sekä käydään läpi monilähetysten teoriaa. Luvussa kolme esitetään MICS-järjestelmän toiminta. Luvussa neljä esitetään TMR-monilähetysjärjestelmän toteutus, sekä arvioidaan toteutetun monilähetysjärjestelmän toiminta. Luvussa viisi esitetään työn päätelmät, sekä ehdotukset jatkokehitystä varten.

Luku 2

Reititys- ja monilähetysprotokollat

Tässä luvussa käydään läpi eri verkkoympäristöihin suunniteltuja monilähetysprotokollia. Monilähetyspalvelut ovat tarpeellisia monille sovelluksille, kuten videokonferenssipalvelut ja IP-puhepalvelut. Tämän vuoksi monilähetysprotokollia on tutkittu jo pitkään ja ratkaisuja monille monilähetysviestivälityksessä yleisille ongelmille on jo ehditty tutkia.

2.1 Monilähetys IP-verkoissa

Tunnetuin IP-verkoissa käytetyistä monilähetysreititysprotokollista on Protocol Independent Multicast, suomeksi protokollariippumaton monilähetys. Protokollan nimen mukaisesti PIM-monilähetysprotokollat eivät ole riippuvaisia yksittäisestä täsmälähetysreititysprotokollasta, vaan ne voivat toimia useiden eri täsmälähetysreititysprotokollien kanssa. Tunnetuimmat PIM-protokollat ovat PIM-SM- (eng. Protocol Independent Multicast - Sparse Mode) ja PIM-DM-protokollat (eng. Protocol Independent Multicast - Dense Mode).

Protocol Independent Multicast - Sparse Mode

PIM-SM-protokolla (eng. Protocol Independent Multicast - Sparse Mode) on suosittu monilähetysprotokolla IP-verkoissa [27]. PIM-SM-protokolla kuvataan Internet-standardissa RFC 4601 [8]. PIM-SM-protokollalla voidaan toteuttaa monilähetysreititys tehokkaasti monilähetysryhmille, joissa ryhmän jäsenet ovat hajautuneet jopa useiden WAN-verkkojen (eng. Wide Area Network) alueelle. PIM-SM-protokollassa monilähetysryhmän vastaanotta-

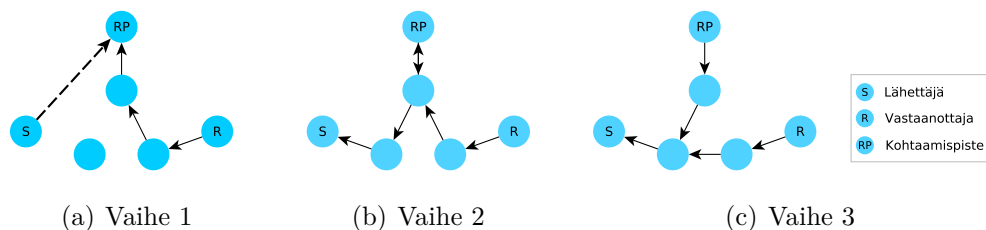
jien ei tarvitse tietää lähettäjiä ennalta, eikä lähettäjien tarvitse tietää vastaanottajia ennalta.

PIM-SM-protokollassa valitaan yksi verkon monilähetyksreititin toimimaan kohtaamispisteenä (eng. Rendezvous Point). Uudet monilähetyksryhmän jäsenet lähettävät liittymisviestinsä kohtaamispisteelle ja ryhmään lähettävät solmut lähettävät monilähetyspaketit kohtaamispisteelle.

PIM-SM-protokollassa ylläpidetään reititystaulua, jota kutsutaan nimellä MRIB-reititystaulu (eng. Multicast Routing Information Base). MRIB-reititystaulu voidaan selvittää suoraan täsmälähetyksprotokollan reititystiedosta, tai jostain erityisesti monilähetykseen tarkoitetun reitityksprotokollan reititystiedosta. MRIB-reititystaulun avulla selvitetään mille solmulle liittymis- ja karsimisviestit tulee lähettää.

Koska liittymisviestit lähetetään kohti monilähetyksdatan vastaanottajaa, kuvaa MRIB-reititystaulu monilähetyksdatan paluupolun. Täsmälähetyksprotokollien reititystietokannassa (eng. Routing Information Base, RIB) kuvataan solmu jolle vastaanotettu paketti tulee reitittää. Täsmälähetyksprotokollien RIB-reititystietokantaan verrattuna MRIB-reititystietokanta kuvaa siis miltä solmulta monilähetyksdata otetaan vastaan, eli datan paluupolun, kun taas RIB-reititystietokannassa kuvataan mille solmulle vastaanotettu paketti välittää, eli paketin edelleenvälityspolun.

PIM-SM-protokollan toiminta jakautuu kolmeen vaiheeseen. Vaiheet on esitetty kuvassa 2.1. Kuvassa täsmälähetykset on merkitty katkoviivanuolilla ja ryhmätillaukset yhtenäisillä nuolilla.



Kuva 2.1: PIM-SM monilähetyksprotokollan kolme vaihetta. Katkoviiva kuvaa täsmälähetyksä ja jatkuva nuoli voimassa olevaa ryhmätillauksa.

Ensimmäisessä vaiheessa muodostetaan kaikkien vastaanottajien kesken jaettu monilähetyksreitityspuu. Ryhmän jäseneksi liittyvä solmu lähettää liittymisviestin paikalliselle monilähetyksreitittimelleen. Paikallinen reititin tilaa itselleen kyseisen ryhmän seuraavalta reitittimeltä kohti kohtaamispistettä. Liittymisviestit kulkevat hyppy hypyltä (eng. hop by hop) kohti kohtaamis-

pistettä, kunnes kohtaamispiste saavutetaan, tai saavutaan reitittimelle joka on jo tilannut kyseisen ryhmän itseään seuraavalta reitittimeltään. Kaikkien vastaanottajien tilattua ryhmän, on ryhmän levityspuu samanlainen kaikille vastaanottajille, puun juuren alkaessa kohtaamispisteeltä ja lehtien päättyessä ryhmän vastaanottajien paikallisiin reitittimiin. Monilähetyksdatan lähettäjä lähettää datan paikalliselle reitittimelleen, joka täsmälähettää sen edelleen kohtaamispisteelle, mistä data välitetään jaettua monilähetyksreitityspuuta pitkin vastaanottajille.

Vaihe 1 on esitetty kuvassa 2.1(a). Lähettävä solmu S lähettää monilähetyspaketit täsmälähetyksenä kohtaamispisteelle. Täsmälähetyks on esitetty kuvassa katkoviivalla. Solmu R tilaa monilähetyksryhmän kohtaamispisteeltä. Monilähetyksatilaukset on kuvattu nuolilla, jotka etenevät solmulta R kohtaamispisteelle. Tilauksen edettyä kohtaamispisteelle, verkolla on jaettu monilähetyspuu, jossa on vain yksi haara, joka kulkee kohtaamispisteeltä RP solmulle R. Solmun S kohtaamispisteelle RP täsmälähetyksenä välittämät paketit lähetetään edelleen monilähetyksenä solmulle R.

Ensimmäisen vaiheen jälkeisessä tilassa verkon resursseja ei käytetä optimaalisesti, koska kohtaamispiste saattaa olla epäedullisesti sijoittunut vastaanottajan tai lähettäjien kannalta. Lisäksi lähettäjän paikalliselle reitittimelle ja kohtaamispisteelle aiheutuu ylimääräistä kuormaa monilähetyksien enkapsuloinnista täsmälähetykseksi.

PIM-SM-protokollan toisessa vaiheessa kohtaamispiste muodostaa lähettäjakohtaisen levityspuutilan kohti lähettäjää. Lähettäjakohtainen levityspuu muodostetaan lähettämällä liittymisviesti kohtaamispisteeltä kohti lähettäjää. Kun reitityspuu on muodostunut, vastaanottaa kohtaamispiste monilähetykspaketit monilähetyksenä lähettäjältä. Tällöin monilähetyksdatan lähettäminen täsmälähetyksenä kohtaamispisteelle voidaan lopettaa.

Vaihe 2 on esitetty kuvassa 2.1(b). Kohtaamispiste RP tilaa monilähetyksryhmän suoraan lähettäjäsolmulta S. Tilauksen edettyä lähettäjäsolmulle S, kohtaamispiste RP vastaanottaa lähettäjäsolmun S viestit monilähetyksenä, joten erillinen täsmälähetyks solmujen välillä voidaan lopettaa. Vaiheen 2 jälkeen kohtaamispisteeltä RP on lähdekohtainen monilähetyksreitityspuu lähettäjäsolmulle S.

Toisen vaiheen jälkeisessä tilassa lähettäjäsolmun ei tarvitse lähettää monilähetykspaketteja täsmälähetyksenä kohtaamispisteelle, joten lähetyksessä vältetään turhalta enkapsuloinnilta. Monilähetyks välitetään kuitenkin edelleen kohtaamispisteen kautta, joka voi olla epäedullista solmujen sijainnista riippuen.

PIM-SM-protokollan kolmannessa vaiheessa monilähetyksen vastaanottaja muodostaa lähdekohtaisen monilähetystreitityspuun. Lähdekohtainen monilähetystreitityspuu muodostetaan lähettämällä tilaus kohdesolmulta kohti lähettäjä. Kun lähdekohtainen monilähetystreitityspuutila on muodostettu, vastaanottajalle saapuu lähettäjän monilähetytsdata sekä suoraan lähettäjältä, että kohtaamispisteeltä. Jotta samaa monilähetytsdataa ei vastaanotettaisi kahdesti, vastaanottaja peruu ryhmän tilauksen kohtaamispisteeltä kyseisen lähettäjän osalta.

Vaihe 3 on esitetty kuvassa 2.1(c). Kohdesolmu R tilaa monilähetyksryhmän suoraan lähettäjäsolmulta S. Kun tilaus on edennyt lähettäjäsolmulle S ja kohdesolmu vastaanottaa monilähetytsdataa suoraan lähettäjäsolmulta, tilaus kohtaamispisteelle RP voidaan poistaa. Vaiheen 3 jälkeen kaikilla vastaanottajilla, eli solmulla R, on lähdekohtainen ryhmälähetystreitityspuu lähettäjäsolmulle S.

Protocol Independent Multicast - Dense Mode

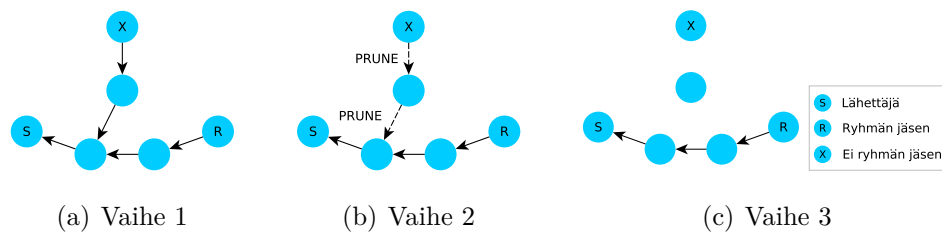
PIM-DM-protokolla (Protocol Independent Multicast - Dense Mode) on monilähetyksprotokolla verkkoihin, joissa lähes kaikki solmut tilaavat lähetettävän monilähetyksryhmän. PIM-DM on kuvattu Internet-standardissa RFC 3973 [2].

PIM-DM-protokollassa lähetyksen alkaessa oletetaan, että monilähetyksryhmän viestit tulee lähettää kaikille monilähetystreitittimille. Vasta lähetyksen vastaanotettuaan reititin tarkistaa onko sillä yhtään monilähetysvastaanottajaa, tai monilähetystreititintä jolle sen tulisi reitittää lähetys edelleen. Jos ei ole, se lähettää karsimisviestin edelliselle reitittimelle, joka lopettaa monilähetyksen kyseiselle reitittimelle ja suorittaa saman tarkistuksen. Jos vastaanottajia ei ole, peruu reititin tilauksen itseään edeltävältä reitittimeltä.

Protokollan toiminta on esitetty kuvassa 2.2. Ensimmäisessä vaiheessa solmu S on lähettäjä ja sen lähettämät paketit reititetään solmuille X ja R, solmun R ollessa ainoa ryhmälähetyksen tilaaja. Toisessa vaiheessa solmu X lähettää karsimisviestin, koska sillä ei ole tilaajia ryhmälle. Myös solmun X seuraava reititin lähettää karsimisviestin, koska sillä ei ole naapurinaan yhtään reititintä, joka tilaisi ryhmälähetystä. Vaiheessa 3 ryhmälähetysviestit lähetetään vain solmulle R.

Viestien tulvittamista kaikille ja tulvituksen lopettamista karsimisviesteillä kutsutaan tulvita ja karsi -kierroksi (eng. flood and prune cycle).

Ryhmän tilaamisen peruminen ei ole lopullinen, vaan peruminen vanhenee tietyn ajan jälkeen. Tilan vanhentuminen aiheuttaisi ylimääräisen tulvita ja karsi -kierroksen. Ylimääräisen kierroksen välttämiseksi lähettäjä lähettää



Kuva 2.2: PIM-DM monilähetysprotokollan kolme vaihetta. Jatkuva viiva esittää voimassaolevaa ryhmätilausta. Karsimisviestit on esitetty katkoviivalla, joiden rinnalla lukee PRUNE (suom. karsi).

erityisen tilanpäivitysviestin (eng. state update message) tasaisin väliajoin, joka välitetään monilähetyspuussa alaspäin kaikille reitittimille. Päivitysviesti aiheuttaa tilauksien perumisen uusimisen, eli karsimisviestien lähettämisen ryhmille joilla ei ole tilaajia. Tilauksien karsimiset siis uusitaan, jolloin tilauksen peruminen ei vanhene ja erillistä tulvita ja karsi kierrosta ei tarvita.

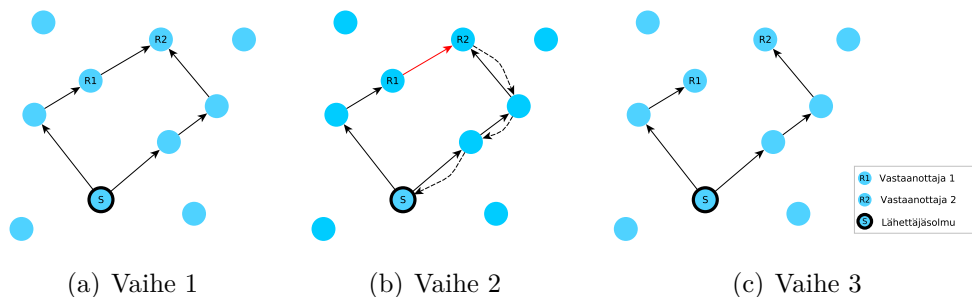
PIM-DM-protokollan hyöty PIM-SM-protokollaan verrattuna on, ettei se vaadi keskitettyä kohtaamispistettä, joka on kriittinen piste protokollan toiminnan kannalta (eng. single point of failure). Lisäksi PIM-DM voi olla parempi vaihtoehto verkkoihin, joissa suurin osa verkon jäsenistä tilaa monilähetysryhmän, koska tilauksia ei tarvitse tehdä erikseen, vaan monilähetysryhmä lähetetään oletuksena kaikille reitittimille, ylimääräisten reitittimien perueissa monilähetyksen erikseen.

Paluupolkuvälitys

Molemmat protokollat, sekä PIM-DM, että PIM-SM, hyödyntävät paluupolkuvälitystä (eng. Reverse Path Forwarding, RPF) kierronestotekniikkana ja se esitellään molempien protokollien Internet-standardeissa[8][2]. Paluupolkuvälityksessä tarkistetaan vastaanotettiin monilähetyspaketti samalta väylältä, jolle se reititettäisiin lähettäjäsolmulle lähetettäessä. Jos paketti vastaanotettiin väärältä väylältä, se hylätään ja lähettäneelle reitittimelle välitetään karsimisviesti.

Kuvassa 2.3 esitetään esimerkki paluupolkuvälityksen toiminnasta. Ryhmälä on vastaanottajat R1 ja R2, joista vastaanottaja R2 saa ryhmän viestit kahta reittiä. Kun solmu R2 vastaanottaa lähettäjän S paketin, tarkistaa se mitä kautta paketti vastaanotettiin. Tarkistus kuvataan vaiheessa 2. Kuvassa paluupolku, eli reitti jota pitkin paketti reititettäisiin lähettäjälle, on kuvattu katkoviivalla. Jos paketti vastaanotettiin solmulta R1, paluupolkuvälitystarkistus epäonnistuu ja tilaus solmulta R1 perutaan. Vaihe 3 esittää

lopputilaa, jossa molemmat jäsenet vastaanottavat monilähetyspaketit vain yhtä reittiä pitkin.



Kuva 2.3: Paluupolkuvälityksen toiminta

Paluupolkuvälitys olettaa lähettäjän ja reitittimen välisen reitin symmetriseksi. Oletus on seurausta siitä, että vastaanotetun paketin väylän oikeellisuus tarkistetaan lähetettävien pakettien reititystiedon avulla. PIM-protokollat eivät siis kykene hyödyntämään epäsymmetrisiä täsmäreititysprotokollia. Epäsymmetrisiä linkkejä esiintyy esimerkiksi verkoissa, joissa hyödynnetään satelliittilinkkejä[10].

2.2 Reititys MANET-verkoissa

MANET-verkoilla tarkoitetaan tiettyyn tarkoitukseen (ad hoc) rakennettuja verkkoja, joissa solmut ovat jatkuvassa liikkeessä. Lisäksi solmuilla saattaa olla rajalliset resurssit, kuten esimerkiksi rajallinen sähköenergia paristokäyttöisillä solmuilla.

MANET-verkkojen tutkimus on kasvanut huomattavasti 1990-luvulta lähtien. MANET-verkot ovat ajankohtaisia, koska langattomaan kommunikointiin kykenevien laitteiden määrä ja laatu on kasvussa. Laitteita, jotka kykenevät kommunikoimaan langattomasti keskenään, ovat esimerkiksi kännykät ja kannettavat tietokoneet.

MANET-verkkoihin lasketaan myös sensoriverkot. Sensoriverkoilla tarkoitetaan verkkoja joissa tyypillisesti paristokäyttöiset solmut mittaavat antureilla ympäristöään ja lähettävät mittaustiedon keskitettyyn sijaintiin prosessoitavaksi. Sensoriverkon solmut muodostavat keskenään verkon, jonka kautta mittaustietoa voidaan välittää. Tyypillisesti yksi tai useampi verkon solmu on yhteydessä ulkoiseen verkkoon, jonka kautta mittaustieto voidaan välittää keskitettyyn sijaintiin prosessoitavaksi.

MANET-verkoille tyypillistä on verkon solmujen liike, joka aiheuttaa topologiamuutoksia verkkoon. Perinteisissä IP-verkoissa topologiamuutosten määrä oletetaan pieneksi, koska verkot ovat yleensä rakennettu langallisista yhteyksistä. MANET-verkoissa topologiamuutokset ovat kuitenkin jatkuvia ja käytetyn reititysprotokollan tulee ottaa ne huomioon.

2.2.1 Täsmälähetys

Tässä luvussa käydään läpi MANET-verkkojen täsmäreititysprotokollia. MANET-verkkoihin on kehitetty useita täsmäreititysprotokollia, jotka voidaan karkeasti jakaa seuraaviin kategorioihin: proaktiiviset, reaktiiviset, hierarkiset ja geograafiset reititysprotokollat.

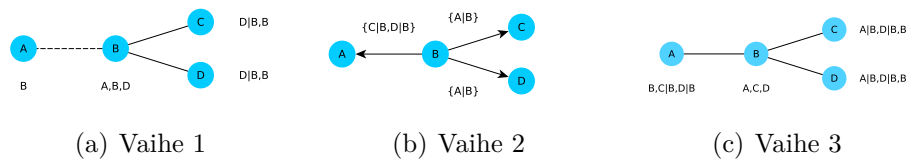
Proaktiivinen reititys

Proaktiivisissa protokollissa reitit kohteisiin selvitetään ennen varsinaista lähetystä. Reititystietoa ylläpidetään erillisessä reititystaulussa, jossa esitetään minne eri kohteille välitettävät paketit tulee välittää. Reititystaulu tulee olla selvillä ennen lähetystä ja se voidaan selvittää staattisesti käyttäjän antaman tiedon perusteella, tai dynaamisesti signaloimalla naapuri- ja reititystietoa verkon yli. Proaktiivisia reititysprotokollia ovat esimerkiksi DSDV [24] ja OLSR [6]. Proaktiivisessa reitityksessä itse paketinvälitys on nopeaa, koska reititystieto on selvillä ennen paketin välittämistä. Reititystiedon ylläpitäminen verkoissa, joissa solmut liikkuvat paljon on kuitenkin haastava tehtävä ja solmujen liikkeessa reititystietojen päivittäminen verkon yli saattaa kuluttaa suuren osan verkon siirtokapasiteetista.

Kuvassa 2.4 on esimerkki proaktiivisesta reitityksestä uuden linkin syntyessä verkkoon. Vaiheessa 1 linkki syntyy solmujen A ja B välille. Solmujen A ja B reititystauluissa näkyy suora naapuruus molempien solmujen osalta, mutta solmuilla C ja D ja solmulla A ei ole reititystietoa toisistaan. Vaiheessa 2 solmu B lähettää tiedon havaitsemastaan uudesta reitistä solmuille C ja D, sekä oman reititystaulunsa sisällön solmulle A. Vaiheessa 3 kaikilla verkon solmuilla on täysi reititystieto verkosta, eli verkko on konvergoitunut.

Optimized Link State Routing Protocol

Optimized Link State Protocol (OLSR) on määritetty Internet-standardissa RFC 3626 [6]. OLSR on proaktiivinen mobiileihin langattomiin verkkoihin optimoitu versio perinteisestä linkkitilareitityksestä. OLSR-protokollan keskeinen konsepti on MPR-solmujen joukko (eng. Multipoint Relay Set). Linkkitilaprotokollille ominainen linkkitilaviestien tulvittaminen voidaan OLSR-protokollassa suorittaa vähäisemmällä viestinvälityksellä tulvittamalla link-



Kuva 2.4: Uuden linkin syntyminen proaktiivisessa reitityksessä. Katkoviiva esittää syntyvää linkkiä. Vaiheissa 1 ja 3 solmujen vieressä lukee solmujen reititystaulu ja suorat naapurit. Vaiheessa 2 nuolet esittävät etäisyysvektorilähetystä ja nuolen rinnalla lukee etäisyysvektorin sisältö.

kitilätiedot ainoastaan MPR-joukon solmujen välillä.

MPR-solmujen joukko selvitetään OLSR-protokollassa hajautetusti. Solmut signaloivat toisilleen hajautetusti naapurustotietonsa, jonka avulla solmut kykenevät selvittämään likimain optimaalisen MPR-solmujen joukon. OLSR-protokollan MPR-joukon valinta-algoritmi valitsee MPR-solmut siten, että MPR-joukko ei ole jaettu.

OLSR-protokollan naapuruston selvittämiseen käyttämä protokolla esitetään OLSR-protokollan Internet-standardissa, mutta se on myös eriytetty omaksi standardikseen RFC 6130 [5]. OLSR-protokollan naapuruston selvittämiseen käytettyyn algoritmiin palataan myöhemmin tässä työssä.

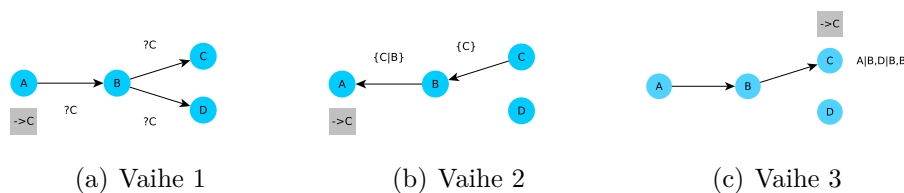
OLSR siis optimoi signaalitiedon tulvitusta tulvittamalla linkkitilaviestit MPR-solmujen välillä, eikä kaikkien solmujen kesken. Toisena optimointina OLSR vähentää lähetettyjen linkkitilaviestien määrää tulvittamalla verkkoon ainoastaan MPR-solmujen linkkitilätiedon. Kaikkien solmujen linkkitilätietoa ei siis tulviteta verkkoon, jolloin tiheissä verkoissa välitettyjen viestien määrä on perinteisiin linkkitilaprotokolliin verrattuna huomattavasti pienempi.

Kolmantena optimointina OLSR tulvittaa verkkoon ainoastaan MPR-solmulta sen MPR-valitsijoiden solmuille liittyvät linkit. Perinteisistä linkkitilaprotokollista poiketen verkkoon ei siis mainosteta kaikkia verkon linkkejä, vaan verkkoon välitetään vain osittainen tieto niin, että MPR-solmuilla on ainakin yksi reitti MPR-valitsijasolmuilleen. Verkon solmujen reitittäessä paketteja verkossa, reitit kulkevat siis MPR-solmujen kautta, koska ainoastaan MPR-solmut osaavat reitittää paketteja omille MPR-valitsijasolmuilleen.

Koska OLSR-protokollan tehokkuus perinteisiin linkkitilaprotokolliin verrattuna perustuu ennen kaikkea MPR-joukkojen hyödyntämiseen, on protokolla sitä tehokkaampi mitä tiheämpi verkko on. Mitä tiheämpi verkko on, sitä pienempi määrä MPR-solmuja voi kattaa koko verkon.

Reaktiivinen reititys

Reaktiivisessa reititysprotokollassa reitti kohteeseen selvitetään vasta lähetysten yhteydessä. Tunnettu ja hyvin määritelty reaktiivinen reititysprotokolla on AODV (eng. Ad hoc On-Demand Distance Vector Routing), joka on määritelty Internet-standardissa RFC 3561 [23]. Koska reititystietoa ei ylläpidetä erikseen, ei reititystä varten tarvita signalointia pääsääntöisesti muuten kuin pakettia lähetettäessä. Lähetysten yhteydessä solmu yleislähetää verkkoon reititystietopyynnön kohteelle. Pyyntöön vastaa joko kohdesolmu, tai solmu jolla on talletettuna reitti kyseiselle kohteelle. Reititystietovastauksen edetessä verkossa kohti pyynnön tekijää, verkon solmut tallettavat reititystiedon muistiin mahdollisia tulevia pyyntöjä tai omia lähetyksiään varten. Tyypillisessä tapauksessa reitin pyytjä saa vastauksena joukon reittejä, joista se voi valita parhaan. Reaktiivisen reititysprotokollan hyötynä on, ettei reititystietoa tarvitse ylläpitää niiden solmujen osalta, jotka eivät ota osaa kommunikointiin. Haittana on lisääntynyt viive lähetettäessä kohteelle jolle ei ole valmista reittiä tiedossa.



Kuva 2.5: Esimerkki reaktiivisesta reitityksestä.

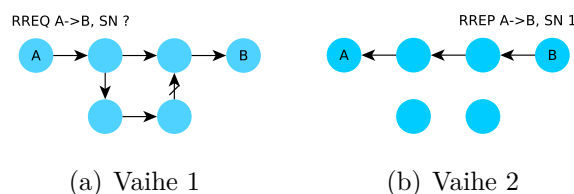
Kuvassa 2.5 on esimerkki reaktiivisesta reitityksestä. Vaiheessa 1 solmu A vastaanottaa solmulle C lähetettävän paketin. Solmulla A ei ole reittiä solmulle C, joten se yleislähetää reitipyynnön solmusta C. Vaiheessa 2 solmu C vastaa reitituskyselyyn ja reitituskysely välitetään solmulle A ja vaiheessa 3 paketti välitetään solmulle C.

Ad hoc On Demand Distance Vector Protocol

Ad hoc On Demand Distance Vector -protokolla (AODV) on määritetty Internet-standardissa RFC 3561 [23]. Se on reaktiivinen reititysprotokolla, eli reitti kohteeseen selvitetään paketin saapuessa välitettäväksi. AODV käyttää reititykseen etäisyysvektoreita, eli reititettäessä välitetään tietoa solmujen etäisyyksiä toisistaan. AODV käyttää kohdesarjanumeroita välttääkseen reitityssilmukat ja etäisyysvektoriprotokollille tyypillisen äärettömään laskeamisen ongelman (eng. counting to infinity).

Kuvassa 2.6 on esitetty esimerkki AODV-reitityksestä. Vaiheessa 1 solmul-

la A on paketti reititettävänä solmulle B. Koska solmulla A ei ole muistissaan reittiä solmulle B, yleislähetää se reittihakupaketin (RREQ, eng. route request) verkkoon. Koska solmulla A ei ole tiedossa sarjanumeroa (SN, eng. serial number) tälle reitille, merkitsee se pyynnön sarjanumeron tuntemattomaksi. Reittihaku etenee verkossa solmu solmulta. Verkon solmut tallettavat reitin RREQ-paketin edelliselle hypylle, jotta ne osaavat reitittää takaisin mahdollisen reititysvastauspaketin (RREP, eng. route reply). Kuvan verkon alareunaa etenevä RREQ-paketti hylätään B:n naapurisolmulla, koska se on vastaanottanut verkon yläreunaa edenneen RREQ-paketin aiemmin. Jos jollain verkon solmulla olisi valmis reitti solmulle B, vastaisi se suoraan RREQ-pakettiin, välittämättä RREQ-pakettia eteenpäin. Verkon muilla solmuilla ei kuitenkaan ole kyseistä reittiä, joten RREQ-paketti välittyy kohteelle, eli solmulle B, asti.



Kuva 2.6: AODV-protokollan reittipyyntö.

Vaiheessa 2 solmu B muodostaa RREP-paketin ja lähettää sen vastauksena RREQ-pakettiin. Tämä on ensimmäinen reittipyyntö solmulle B, joten se asettaa RREP-paketin sarjanumeroksi numeron 1. RREP-paketti välittää RREQ-paketin välittäneiden solmujen tallettamaa paluureittiä pitkin. Vaiheessa 1 verkon alareunaa edennyt RREQ-paketti hylättiin, joten vastaus lähetetään ainoastaan verkon ylälaidan paluureittiä pitkin. Vastauspaketissa on mukana vastauksen kulkemien hyppyjen määrä, joten jos solmu A vastaanottaisi useamman RREP-paketin, osaisi se valita mahdollisista reiteistä lyhyimmän.

Verkon solmut tarkkailevat naapuriyhteyksiensä tilaa. Jos yhteys johonkin naapuriin katoaa, saattaisi muutoksesta tietämätön solmu reitittää paketin toimimattoman linkin yli. Jotta paketteja ei reititetäisi katkenneen linkin yli, solmut pitävät kirjaa naapurisolmuista, jotka saattavat olla kiinnostuneita naapurilinkin tilasta. Jos linkki katoaa, ilmoitetaan muutoksesta naapureille, joiden reititykseen muutoksella saattaisi olla vaikutusta. Linkin katkeaminen ilmoitetaan reititysvirhepaketilla (RERR-paketti, eng. route error packet). RERR-paketin levitessä verkossa, verkon solmut merkitsevät linkkiä käyttäneet reitit toimimattomaksi.

Jos toimimattomaksi merkittyä reittiä tarvitaan myöhemmin, muodostaa lähettäjä uuden RREQ-paketin. Uuden RREQ-paketin sarjanumero merkitään yhtä suuremmaksi kuin aikaisemman, toimimattomaksi merkityn, reitin sarjanumero. RREQ-paketin edetessä verkossa, verkon solmut, joilla on valmiita reittejä kyseiselle kohdesolmulle, tarkistavat että talletetun reitin sarjanumero on yhtä suuri tai suurempi kuin RREQ-paketin. Näin estetään vanhojen toimimattomien reittien uudelleenkäyttäminen. Kohdesolmun vastaanottaessa RREQ-paketin, asettaa se oman sarjanumeronsa RREQ-paketin sarjanumeron ja oman sarjanumeronsa maksimiksi. Näin solmun RREQ-paketteihin antama sarjanumero pysyy jatkuvasti kasvavana.

AODV-protokollan standardissa ei kuvata tarkasti kuinka monilähetys tulisi toteuttaa. Protokollassa kuvataan kuitenkin paketeille monilähetykselle erityisiä pakettioptioita ja ohjeistetaan, että monilähetysprosessointia osaa-mattomien solmujen tulisi prosessoida monilähetys-IP-osoitteisiin lähetetyt paketit kuten normaalit täsmälähetyspaketit. Vaikka standardi ei sisällä monilähetystoiminnallisuutta, on AODV-protokollalle esitetty monilähetystoiminnallisuus erikseen [29]. AODV-protokollan monilähetysprotokolla esitellään seuraavassa luvussa.

Hierarkinen reititys

Hierarkisessa reititysprotokollassa solmut jaetaan hierarkkisiin ryhmiin. Reititys ryhmien sisällä ja niiden välillä suoritetaan erikseen. Tunnettu esimerkiksi hierarkkisesti jaetusta verkosta ovat Internet-verkot, joissa korkeimmalla tasolla verkot on jaettu autonomisiin järjestelmiin joiden välillä reititys suoritetaan yleensä BGP-protokollalla (eng. Border Gateway Protocol) [25] ja verkkojen sisällä esimerkiksi RIPv2-protokollalla (eng. Routing Information Protocol version 2) [18] tai OSPF-protokollalla (eng. Open Shortest Path First) [20]. MANET-reitityksessä hyvä esimerkki hierarkkisesta reitityksestä on ZRP (eng. Zone Routing Protocol) [11]. ZRP-protokollassa solmut on jaettu ryhmiin joiden sisällä käytetään proaktiivista reititystä ja ryhmien välillä reaktiivista reititystä. ZRP-protokolla on siis yhdistelmä proaktiivisesta ja reaktiivisesta reitityksestä.

Geograafinen reititys

Geograafisissa reititysprotokollissa reitityksessä käytetään hyödyksi solmujen sijaintitietoa. Reitityksessä ei ylläpidetä erillistä reititystietoa verkon solmuille, vaan paketti välitetään paikkatiedon perusteella hypy hypyltä lähemmäs kohdetta. Yksinkertaisimmillaan paketti välitetään ahneesti (eng. greedy) kohti vastaanottajaa, jolloin vastaanottajaksi valitaan aina kohdetta lähimpänä sijaitseva naapuri. Tällainen reititys saattaa kuitenkin epäonnistua, jos kohteeseen ei ole reittiä kuljettaessa joka hypyllä lähemmäs kohdet-

ta, vaan välissä tulisi edetä pois päin kohteesta. Näitä erikoistilanteita varten on kehitetty algoritmeja joita on esitetty lähteessä [30]. Koska erillistä reititystietoa ei ylläpidetä, geograafinen reititys toimii lähes tilattomasti ja se skaalautuu hyvin isommillekin ad hoc -verkoille [30]. Geograafisessa reitityksessä tulee solmujen tuntee oma sijaintinsa, joka voidaan selvittää käyttäen esimerkiksi GPS-paikannusta (eng. Global Positioning System).

2.2.2 Monilähetys

Tässä luvussa käsitellään MANET-verkkoihin suunniteltuja monilähetysprotokollia. MANET-verkkoihin suunniteltujen monilähetysprotokollien tulee kyetä selviytymään topologiamuutoksista ja niiden tulee hyödyntää langattoman siirtotien edut langalliseen siirtotiehen verrattuna.

Ad hoc On Demand Distance Vector Routing Protocol Monilähetys

AODV-protokolla itsessään ei tue monilähetystä, mutta sen päälle on suunniteltu erillinen monilähetysprotokolla, MAODV-protokolla (eng. Multicast Ad Hoc Distance Vector) [29]. MAODV-protokolla mahdollistaa monilähetysryhmien muodostamisen tarpeen vaatiessa, ilman aikaisempaa tietoa (eng. on-demand).

Kuten AODV täsmälähetysprotokollassa solmut ylläpitävät sarjanumeroa reitityspyyntöjä varten, myös monilähetyksessä hyväksikäytetään sarjanumeroita. Jokaisella levitysryhmällä on ryhmänjohtaja, joka ylläpitää ryhmän sarjanumeroa.

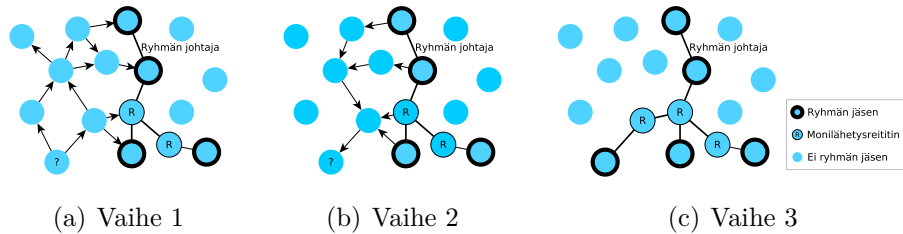
Solmun liittyessä monilähetysryhmään, se tarkistaa onko tiedossa valmista reittiä jollekin ryhmään kuuluvalla solmulla. Jos on, se täsmälähetää RREQ-paketin kyseiselle solmulle. RREQ-paketin J-lippu (eng. JOIN-flag) tulee olla aktivoitu. Jos valmista reittiä ei ole tiedossa, solmu yleislähetää RREQ-paketin verkkoon.

Täsmälähetyksissä solmu, jolla on tiedossa reitti kohteeseen, voi vastata RREQ-pakettiin. Monilähetyksessä solmun ei kuitenkaan tule vastata RREQ-pakettiin, jos se ei ole monilähetysryhmän jäsen, tai sen reitin.

RREQ-paketti lähetetään verkkoon useamman kerran ja jos siihen ei määritettyyn aikaan mennessä tule vastausta, oletetaan ettei ryhmällä ole vielä muita jäseniä. Tässä tapauksessa solmusta tulee ryhmän johtaja ja se alustaa ryhmän sarjanumeron arvoon 1.

Kuvassa 2.7 esitetään protokollan toiminta solmun liittyessä monilähetysryhmään. Vaiheessa 1 solmu, jolla ei ole valmista reititystietoa ryhmälle, yleislä-

hettää RREQ-paketin verkkoon. Solmut uudelleenlähettävät RREQ-paketin, jos ne eivät ole monilähetyspuun jäseniä.



Kuva 2.7: MAODV-protokollan monilähetysreititikysely.

RREQ-paketin vastaanottaessaan solmu tallettaa paluureitin lähettäjälle ja lisäksi lisää paluureitin monilähetysreititystauluun. Monilähetysreititystaulun reitti merkitään kuitenkin epäkelvoksi, ennen kuin kysynyt solmu valitsee kyseisen reitin käytettäväksi reitiksi monilähetysryhmään. Jos solmun kautta kulkeva reitti valitaan, tulee siitä ryhmän reitittävä jäsen ja siten osa monilähetysreitituspuuta.

Monilähetysreitituspuun jäsenet vastaavat RREQ-pakettin RREP-paketilla, johon sisällytetään etäisyys vastanneeseen monilähetyspuun jäseneseen. RREP-paketit välittyvät kyselijälle kuten täsmälähetysprotokollassakin. Kysyvä solmu odottaa määritellyn ajan RREP-pakettien välittymistä ja valitsee sitten käytettävän reitin erityisellä monilähetysreitin aktivointiviestillä (eng. Multicast Activation Message, MACT). RREQ-paketteja välittäneet solmut, joiden reittiä ei valittu, poistavat monilähetysreitin odotettuaan MACT-viestiä määritellyn ajan.

Kuvan 2.7 vaiheessa 2 monilähetyspuun jäsenet vastaavat vaiheen 1 RREQ-pakettiin. RREP-paketit etenevät reititin reitittimeltä kohti kysyjää. Koska RREP-paketit sisältävät etäisyystiedon vastanneeseen levitysryhmän jäseneseen, pystyvät RREP-paketin välittävät solmut valitsemaan parhaan reitin monilähetysryhmään kaikkien RREP-paketin lähettäneistä naapureista. Vaiheen 2 jälkeen kysynyt solmu vastaa MACT-viestillä ja viesti etenee kohti lähintä monilähetyspuun jäsentä, siten että lyhyimmän reitin solmuista tulee reitituspuun jäseniä. Vaiheessa 3 solmu on lisätty ryhmän jäseneksi ja monilähetysviestit välittyvät kaikille ryhmän jäsenille.

Levitysryhmien johtajat yleislähetävät verkkoon hello-viestejä. Jokainen hello-viesti kasvattaa ryhmän sarjanumeroa, joten sen antama tieto päivittää ryhmän aikaisemman tilan. Hello-viestiä käytetään hyödyksi myös jakautuneiden monilähetyspuiden yhdistämisessä.

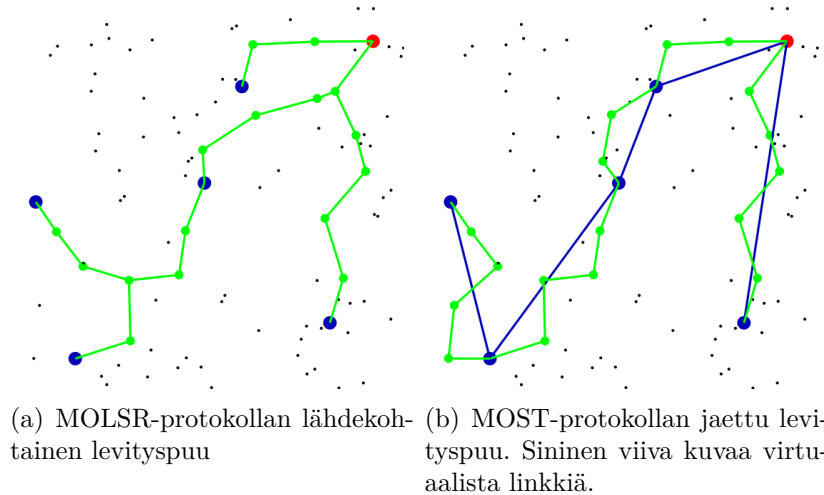
Protokolla sisältää mekanismit linkkien rikkoutumisesta ja palautumisesta selviytymiseen, tilauksen perumiseen ja erottuneiden monilähetyspuiden yhdistämiseen, mutta näiden mekanismien kuvaaminen on jätetty tämän työn ulkopuolelle.

Optimized Link State Routing Protocol Monilähetys

OLSR-protokollalle on esitetty useita monilähetysreititysprotokollia [19], joista tässä esitellään Multicast Overlay Spanning Tree Protocol (MOST) [28] ja Multicast Optimized Link State Routing (MOLSR) [15].

Molemmat, sekä MOST, että MOLSR hyväksikäyttävät OLSR-protokollan optimoitua tulvitusrakennetta jäsenyystiedon välittämiseen. Protokollat eroavat toiminnassaan siinä, että MOLSR-protokollassa rakennetaan lähdekohtainen levityspuu, kun MOST-protokollassa ryhmälle rakennetaan jaettu puu. Lisäksi MOLSR-protokollassa monilähetysreititykseen voi osallistua kaikki monilähetysreititykseen kykenevät solmut, kun MOST-protokollassa reititys suoritetaan ryhmän tilanneiden solmujen kesken.

Kuvassa 2.8 esitetään MOST- ja MOLSR-protokollien muodostamien levityspuiden ero. MOLSR-protokollan muodostama levityspuu on lähettäjäkohtainen, joten paketit välitetään lyhyintä reittiä kaikille vastaanottajille. MOST-protokollan levityspuu muodostetaan vain ryhmän jäsenten kesken ja puu on jaettu. Koska ryhmälähetyspuu on jaettu ryhmän kesken, puu näyttää samalta kaikkien ryhmän jäsenten kannalta.



Kuva 2.8: OLSR-protokollaan perustuvat monilähetysprotokollat MOLSR ja MOST [19].

MOST-protokollassa pakettia ei välitetä yleislähetystenä solmulta toiselle,

vaan ryhmän jäsenet lähettävät paketin täsmälähetyksenä seuraaville vastaanottajille. Tämän vuoksi yhdellä hypyllä vain yksi naapuri voi vastaanottaa paketin. MOLSR-protokollassa paketit yleislähetetään, joten paketti voidaan välittää usealle solmulle yhdellä lähetyksellä.

Vaikka MOST-protokollan tehokkuuden voisi kuvitella kärsivän merkittävästi täsmälähetyksien vuoksi, voi alemman kerroksen MAC-protokollalla olla yllättäviä vaikutuksia tuloksiin [19]. Esimerkiksi IEEE 802.11b-protokollassa täsmälähetykset uudelleenlähetetään automaattisesti, mikä parantaa suorituskapasiteettia (eng. throughput).

Geolähetyksprotokollat

Vastaanottaja saatetaan kohdistaa tietyn solmun sijaan tietyksi geograafiseksi alueeksi, jolloin kaikki tietyn alueen solmut vastaanottavat lähetyksen. Tällainen lähetys on monilähetyksen erityismuoto ja sitä kutsutaan geolähetykseksi (eng. geocast). Geolähetyksprotokollia on useita ja muutamia niistä on kuvailtu ja vertailtu lähteessä [33].

Yksi esimerkki geocast-monilähetyksprotokollista on Location Based Multicast (LBM) [14]. LBM-protokollalle on määritetty kaksi eri toimintatapaa: edelleenvälitysalueisiin perustuva lähetys (engl. LBM-box) ja portaittainen lähetys (engl. LBM-step).

Edelleenvälitysalueisiin perustuva toimintamalli on esitetty kuvaajassa 2.9(a). Tässä toimintamallissa muodostetaan edelleenvälitysalue, joka on pienin suorakulmion muotoinen alue, jonka sisäpuolelle kuuluvat sekä lähettäjä että geolähetyksalue. Paketin vastaanotettuaan solmu edelleenlähettää paketin, jos se on edelleenvälitysalueen sisäpuolella. Paketin saapuessa geolähetyksalueelle, se tulvitetaan kaikille geolähetyksalueen solmuille.

Kuvaajassa 2.9(b) esitetään muunnelma edelleenvälitysalueisiin perustuvasta toimintamallista, jossa edelleenvälitysalue päivitetään jokaisen lähetyksen yhteydessä. Solmu S asettaa edelleenvälitysalueeksi alueen Z ja lähettää paketin. Solmu I vastaanottaa paketin, tarkistaa kuuluvansa edelleenvälitysalueen Z sisälle. Näin ollen solmu I edelleenvälittää paketin, mutta se päivittää lähetyksalueen oman sijaintinsa mukaan niin, että uusi lähetyksalue Z' on pienin suorakulmio, johon sekä solmu I että geolähetyksalue kuuluvat. Seuraava solmu, eli solmu J, suorittaa samat toimenpiteet ja edelleenlähettää paketin edelleenvälitysalueella Z".

Portaittaisessa LBM-lähetyksessä edelleenlähetyksessä ei käytetä edelleenvälitysalueita, vaan vain solmujen sijaintia. Paketin vastaanotettuaan solmu tarkistaa olevansa lähempänä geolähetyksaluetta kuin edellinen solmu ja edelleenvälittää paketin, jos etäisyys oli pienempi. Samoin kuin edelleenväli-

Kuva 2.9: LBM-box edelleenvälitys.

Minimaalista levitysrakenne tulvittamiseen (eng. Minimum Connected Do-

Diagram illustrating a graph structure with 12 nodes. The nodes are arranged in a circular pattern with two central nodes. The nodes are colored light blue, indicating they are not part of the MCDS set. The two central nodes are highlighted with thick black outlines, indicating they are part of the MCDS set.

Legend:

- MCDS-joukon solmu (Node in MCDS set)
- MCDS-joukkoon kuulumaton solmu (Node not in MCDS set)

Kuvassa 2.10 on esitetty esimerkki MCDS-joukon selvittämisestä. Koska kuvan siniset solmut pystyvät yhdessä saavuttamaan kaikki verkon solmut, muodostavat ne verkon optimaalisen levitysrakenteen ja siten optimaalisen MPR-solmujen joukon. Viesti voidaan siis tulvittaa verkossa miltä tahansa solmulta optimaalisesti siten, että se lähetetään tulvitettavaksi lähimmälle MPR-solmulle, eli jommalle kummalle verkon keskiosan solmuista. Tulvitettavan viestin vastaanottanut MPR-solmu tulvittaa sen kaikille naapureilleen ja MPR-joukkoon kuuluvat naapurisolmut tulvittavat viestin edelleen, jolloin saavutetaan kaikki verkon solmut.

SMF-protokollan standardi kuvaa kolme algoritmia minimaalisen levitysrakenteen likimaiseen selvittämiseen, kun tunnetaan naapureiden ja naapureiden naapureiden väliset kaksisuuntaiset linkit [16]. Yksi näistä algoritmeista

on OLSR-protokollan MCDS-joukon valinta-algoritmi. Yksisuuntaisia linkkejä ei SMF-standardissa esitetyissä algoritmeissa oteta huomioon. Tarvittava tieto naapurustosta voidaan selvittää erillisellä naapuruston selvittämiseen tarkoitettulla protokollalla, kuten NHDP (eng. Mobile Ad hoc Network Neighborhood Discovery Protocol), joka esitellään seuraavaksi.

Mobile Ad hoc Network Neighborhood Discovery Protocol

MANET-verkoissa reititysprotokollilla on usein tarve saada tietoa naapurustosta. Naapurustotietoa tarvitaan välittömien naapurien selvittämiseen ja minimaalisen levitysrakenteen laskentaan. OLSR-protokolla [6] ja OSPF-protokollan MPR-joukkoja hyödyntävä versio [3] käyttävät molemmat kahden hypyn naapuritietoa minimaalisen levitysrakenteen laskentaan. Mobile Ad hoc Neighborhood Discovery Protocol (NHDP) [5] mahdollistaa naapuritiedon selvittämisen kahden hypyn päähän. Sen toiminnallisuus on irrotettu OLSR-protokollan standardista. NHDP-protokolla mahdollistaa kaksisuuntaisten linkkien selvittämisen kahden hypyn etäisyydellä, sekä muun lisätiedon jakamisen naapurustoon laajennettavalla tavalla.

2.3 Reititys ja tiedonvälitys viivesietoisissa verkoissa

IP- ja MANET-verkoissa tehdään oletuksia, jotka eivät välttämättä päde kaikissa verkoissa. Oletuksia ovat esimerkiksi, että solmujen välillä on mahdollinen reitti, jota pitkin paketit voidaan välittää ja että maksimi latenssi solmujen välillä ei ole kovin suuri [4] [7]. Joissakin olosuhteissa tällaisten verkkojen rakentaminen ja ylläpitäminen voi kuitenkin olla hankalaa tai jopa mahdottomaa. Esimerkiksi planeettojenvälisissä tietoliikenneyhteyksissä yhteyksien viiveet ovat huomattavasti IP- ja MANET-verkoissa oletettuja viiveitä pitempiä. Lisäksi yhteys saattaa katketa jopa tunneiksi esteiden, kuten toisien planeettojen, vuoksi. Armeijoiden käyttöön suunnitelluissa ad hoc-verkoissa solmujenväliset yhteydet saattavat katketa vihollisen häirinnän, tai solmun tuhoutumisen vuoksi.

Viivesietoisissa verkoissa (engl. Delay Tolerant Network, DTN) ei tehdä IP- ja MANET-verkkojen oletuksia, vaan solmujenväliset yhteydet saattavat katketa ja viiveet voivat olla huomattavan pitkiä. DTN-verkkoihin suunniteltuja reititys ja tiedonsiirtoratkaisuja käydään läpi tässä luvussa.

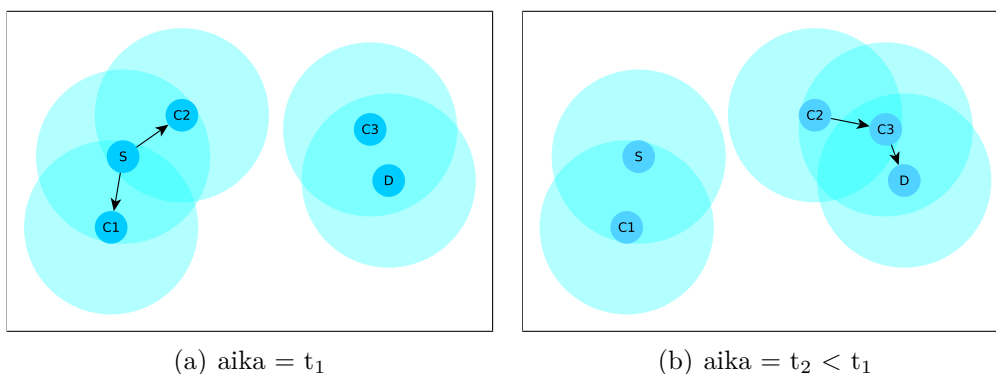
Epideminen Reititys

Vahdat ja Becker esittelivät epidemisen reitityksen (eng. Epidemic Routing)

vuonna 2000 [32]. Epideemisessä reitityksessä lähteen ja kohteen välillä ei oleteta olevan päästä-päähän yhteyttä. Protokollassa paketeista tehdään kopioita muille verkon solmuille, joiden toivotaan kohtaavan kohdesolmu myöhemmässä vaiheessa. Kohdatessaan kohdesolmun, solmu voi toimittaa paketin perille.

Protokolla ottaa myös huomioon resurssienkäytön, siten että viesteille voidaan määritellä TTL-arvo (paketin elinaika, eng. Time-To-Live). TTL arvolle asetetaan kuinka monelle solmulle paketti voidaan kopioida. TTL-arvoa pudotetaan jokaisella kopiolla ja TTL-arvon laskiessa nollaan pakettia ei enää kopioida. Pienellä TTL-arvolla lähetettävät paketit kuluttavat siis vähemmän verkon solmujen resursseja, koska niistä muodostetaan vähemmän kopioita. Kuitenkin, kopioiden määrän ollessa pieni, todennäköisyys, että kohdesolmu kohtaa solmun jolla on kopio viestistä, on pienempi. Mitä suuremmalla todennäköisyydellä kohdesolmun kantoalueelle saapuvalla solmulla on kopio viestistä, sitä nopeammin viesti saadaan toimitettua. Kopioiden määrä, eli TTL, on siis käänteisesti verrannollinen paketin toimitusaikaan.

Kuvassa 2.11 esitetään epideemisen reitityksen toiminta. Hetkellä t_1 lähettäjällä (kuvan solmu S) ei ole suoraa yhteyttä kohdesolmulle R ja se lähettää kaksi kopiota paketistaan solmuille C1 ja C2. Ajanhetkellä T_2 solmu C2 on siirtynyt solmun C3 kantoalueelle ja C2 toimittaa kopion paketista myös solmulle C3. Kohdesolmu R on solmun C3 kantoalueella ja C3 toimittaa paketin kohteelle.



Kuva 2.11: Paketinvälitys epideemisessä reitityksessä [32]

Epideemisessä reitityksessä toistensa kantoalueelle saapuvat solmut synkronoivat pakettikopionsa keskenään. Kuvassa 2.11 ajanhetkellä T_2 solmun C2 kohdatessa solmun C3, solmu C2 lähettää solmulle C3 listan kaikista pakettikopioistaan. Myös C3 lähettää solmulle C2 listan omista pakettikopioistaan. Vastaanotettuaan listan, solmu tarkistaa mitä sille tuntemattomia paketteja

toisella solmulla on ja pyytää niistä kopiot itselleen. Solmut siis synkronoivat pakettikopionsa niin, että molemmilla on samat pakettikopiot kohtaamisen jälkeen.

Bundle-protokolla

Internet standardeissa RFC 4838 ja RFC 5050 kuvataan arkkitehtuuri viivesietoiselle verkoille, sekä viivesietoiisiin verkkoihin tarkoitettu Bundle-protokolla. Bundle-protokolla on on päästä-päähän protokolla viivesietoisille verkoille. Bundle-protokolla on protokollakerroksena reititys-, sekä ohjelmistokerroksen välissä, tarjoten ohjelmistoille toiminnallisuudet viivesietoiisiin verkkoihin. Protokollan standardi ei kuitenkaan määrittele kuinka viivesietoisien verkkojen reitityksen tulee toimia.

Internet-protokollissa on useita oletuksia, jotka estävät niiden tehokkaan toiminnan viivesietoisissa verkoissa:

- Kommunikoivien solmujen välillä on päästä päähän -yhteys istunnon ajan.
- Luotettava tiedonsiirto voidaan toteuttaa tehokkaasti uudelleenlähetyksillä, kun virheestä on saatu ilmoitus vastaanottavalta solmulta.
- Päästä päähän häviö on vähäistä
- Kaikki reitittimet ja solmut tukevat TCP/IP-protokollaa
- Ohjelmistojen ei tarvitse huolehtia tiedonsiirron laadusta

Bundle-protokolla tarjoaa Internet-protokollia luotettavamman tiedonsiirron hyväksikäyttäen viestiä välittävien solmujen tallennustilaa viestinvälityksessä. Näin viesti voidaan välittää seuraavalle solmulle myös tilanteissa, joissa tiedonsiirto seuraavalle solmulle epäonnistuu tai lähettävästä solmusta katkeaa virta tiedonsiirron aikana. Solmulta saattaa katketa virta tiedonsiirron aikana, koska viivesietoisissa verkoissa solmujen väliset yhteydet saattavat olla erityisen hitaita ja katkonaisia.

Viesti välitetään verkossa tallenna ja välitä -periaatteella (eng. store and forward). Viesti poistetaan omasta tallennustilasta vasta kun viesti on saatu välitettäväksi seuraavalle solmulle. Näin viestinvälitys on luotettavaa, vaikka käytettävä tiedonsiirtoverkko olisikin epäluotettava.

Protokollassa on myös mahdollista vaatia viestinvälityksessä erityistä luotettavuutta hallussapito-tiedonsiirron avulla (eng. custody transfer). Tässä

tiedonsiirtomekanismeissa viestin vastaanottava solmu kuittaa edelliselle solmulle erikseen ottavansa vastuun viestin välityksestä. Normaalissa tallenna ja välitä -viestinvälityksessä välittävä solmu luottaa alemman kerroksen ilmoitukseen luotettavasta tiedonsiirrosta, mutta hallussapito-tiedonsiirrosta välittävä solmu odottaa saavansa erillisen bundle-protokollakerroksen kuitauksen seuraavalta solmulta. Viestin uudelleenlähetysvastuu siis siirtyy solmu solmulta verkossa kunnes kohdesolmu on saavutettu.

Vaikka bundle-protokolla ei määrittele kuinka reititys viivesietoisessa verkossa tulee toteuttaa, viivesietoisien verkkojen arkkitehtuurin kuvaava internet standardi antaa kuvauksen eri tyyppisistä kontakteista, joita viivesietoisessa verkossa voi olla. Viivesietoisissa verkoissa saattaa olla joitain kontakteja, jotka ovat jatkuvasti saavutettavissa, kuten DSL-yhteyden takana olevat kontaktit.

Viivesietoisille verkoille erityisiä kontakteja ovat eri tyyppiset katkonaiset kontaktit. Katkonaiset kontaktit saattavat olla saavutettavissa joko täysin satunnaisesti, tietyn aikataulun mukaan, tai ennustettavasti. Täysin satunnainen kontakti voi olla esimerkiksi tiedonsiirtomaston yli lentävä lentokone, joka ilmaisee mastolle olevansa saavutettavissa. Aikataulun mukaan saavutettava kontakti voi olla esimerkiksi satelliitti joka saavuttaa yhteyden muiden solmujen kanssa tarkasti ennustettavan aikataulun mukaan. Ennustettava kontakti voi olla esimerkiksi solmu, jonka todennäköinen reitti ja tulevat saavutettavat yhteydet voidaan ennustaa entisten naapurien ja aikaisemman reitin perusteella.

2.4 Yhteenveto

Tässä työssä toimintaympäristönä on erityisvälitysverkot, eli verkot joissa on tarjolla paremmat resurssit tarjoava runkoverkko ja haasteellisemmat yhteydet verkon laidalla.

Erityisvälitysverkoissa kaikilla esitetyillä monilähetyksprotokollilla on ongelmia toimia tehokkaasti. Erityisvälitysverkkojen katkonaisuuden vuoksi IP-monilähetyksen mekanismit eivät yleisesti ottaen ole toimiva ratkaisu, koska IP-verkoissa oletetaan yhteys lähettävän ja vastaanottavan solmun välillä. Vaikka MANET-verkoissa varaudutaan jatkuviin topologiamuutoksiin, niissä ei varauduta verkon katkonaisuuteen. Viivesietoisissa verkoissa kaikkien solmujen oletetaan olevan katkonaisen yhteyden takana, eikä niissä hyödynnetä mahdollisen runkoverkon parempia tietoliikenneyhteyksiä. Erityisvälitysverkoilla on usein käytettävissään paremmat tietoliikenneyhteydet tarjoa-

va runkoverkko.

Erityisvälitysverkkojen luonteen vuoksi olemassa olevat ratkaisut toimivat niissä huonosti ja tarvitaan erityinen monivälitysjärjestelmä erityisvälitysverkoille. Monilähetysjärjestelmän tulee kyetä välittämään paketit kohteille tehokkaasti, vaikka verkossa olisi väliaikaisia katkoksia ja tiedonsiirtokapasiteetti vähäinen. Lisäksi järjestelmän tulee kyetä hyödyntämään mahdollisen runkoverkon ominaisuuksia.

Luku 3

MICS-järjestelmä

Edellisessä kappaleessa tarkasteltiin reitityksen ja monilähetysreitityksen teoriaa, sekä aikaisempaa tutkimusta. Tässä luvussa esitellään MICS-viestivälitysjärjestelmä (eng. Multi Interface Communications Software), minkä osaksi monilähetyspalvelu toteutetaan. Lisäksi tarkastelemme eri vaihtoehtoja monilähetyspalvelun reititykselle, sekä sovellusarkkitehtuuriselle sijoittamiselle muuhun MICS-järjestelmään nähden.

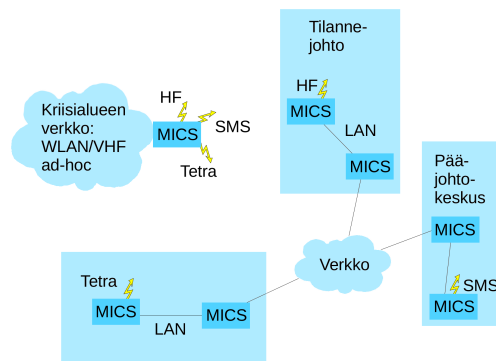
MICS-järjestelmä on viestintäjärjestelmä erityisvälitysverkoille [12]. MICS-järjestelmä on suunniteltu toimimaan viivesietoisten verkkojen kaltaisissa verkoissa joissa viiveet voivat olla pitkiä ja yhteydet katkonaisia. MICS kykenee myös hyödyntämään verkko- ja liityntäteknologioita, jotka eivät tarjoa tarpeeksi hyvää yhteyttä Internet-protokollille [12].

MICS-järjestelmässä on paljon yhtäläisyyksiä viivesietoiisiin verkkoihin. Viivesietoisten verkkojen tapaan MICS-järjestelmä kykenee reitittämään viestit perille katkoksisista huolimatta. Erotuksena viivesietoiisiin järjestelmiin viestistä säilytetään yksittäinen kopio, eikä viestistä välitetä useita kopioita viestinvälityksen varmistamiseksi [12]. Lisäksi reitityksessä MICS-järjestelmän runkoverkossa käytetään IS-IS-protokollan kaltaista reititysprotokollaa, eikä MANET-reititysprotokollia [12].

MICS-järjestelmässä verkko jakautuu kahdenlaisiin solmuihin: runko ja lehtisolmuihin [12]. Runkosolmut ovat yhdistetty toisiinsa paremmilla tiedonsiirtoyhteyksillä, lehtisolmujen kytkeytyessä heikommilla yhteyksillä. Runkoverkon paremmat tiedonsiirtoyhteydet voivat olla esimerkiksi mikroaaltolinkkejä, tai langallisia liityntöjä. Lehtisolmujen yhteydet voivat olla esimerkiksi WLAN, GSM/TETRA, tai VHF-radioyhteyksiä.

Kuvassa 3.1 on esimerkki MICS-verkosta. Käyttöskenaariossa tilanteen johto

on runkoverkossa ja pelastustoimissa käytetty verkko MICS-verkon laidalla. Runkosolmut ovat yhteydessä toisiinsa esimerkiksi Internetin yli hyväksikäyttäen VPN-tekniikkaa (eng. Virtual Private Network) yhteyden suojaamiseksi. Verkon laidalla hyväksikäytetään esimerkiksi HF, SMS ja TETRA-tiedonsiirtoyhteyksiä. Runkoverkossa hyväksikäytetään perinteisiä reititysprotokollia, verkon laidan käyttäessä tulvitusta tai MANET-reititystä.



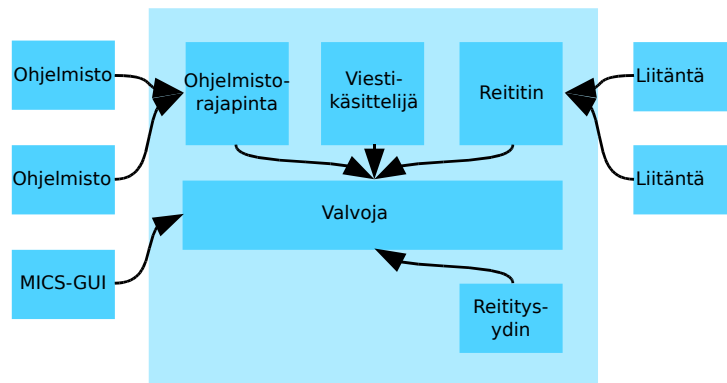
Kuva 3.1: Esimerkki MICS-verkon käyttöskenaariosta [12].

MICS-järjestelmässä itsessään ei ole tukea monilähetykselle. Monilähetystuki on kuitenkin yleisesti tarvittu palvelu erityisvälitysverkoissa, kuten kriisinhallintaverkoissa.

Monilähetystuki voidaan toteuttaa MICS-järjestelmään usealla eri tavalla. Jokaisella toteutustavalla on omat heikkoutensa ja vahvuutensa, jotka on tärkeää huomioida monilähetysjärjestelmää suunniteltaessa. Tässä luvussa käydään läpi MICS-järjestelmän ydinkomponenttien tehtävät ja käytettävissä olevat rajapinnat, jotta voidaan arvioida sopivin toteutustapa monilähetykselle MICS-järjestelmässä.

3.1 MICS-järjestelmän arkkitehtuuri

MICS-järjestelmä on pyritty jakamaan komponentteihin, joilla jokaisella on selkeä tehtävä järjestelmän toiminnassa. Kuvassa 3.2 on esitetty MICS-järjestelmän komponentit. MICS-järjestelmän ydin koostuu viidestä komponentista: ohjelmistorajapinta (eng. Application Interface), viestikäsittelijä (eng. Message Handler), reititin (eng. Router), valvoja (eng. Monitor) ja reititysydin (eng. Routing Core). Jokaisella ydinkomponentilla on oma selkeästi rajattu tehtävä.



Kuva 3.2: MICS-järjestelmän komponentit [13].

Ohjelmistorajapinta on nimensä mukaisesti MICS-järjestelmän rajapinta ohjelmistoille. Se tarjoaa sähköpostirajapinnan viestien lähettämiseen ja vastaanottamiseen. Viestien vastaanottamiseen tarjotaan yleisesti käytetty IMAP-rajapinta (eng. Internet Message Access Protocol) ja lähettämiseen SMTP-rajapinta (eng. Simple Mail Transfer Protocol). Ohjelmistorajapinta välittää ohjelmiston antaman viestin viestikäsittelijälle ja vastaavasti tallettaa viestikäsittelijältä vastaanotetun viestin käyttäjän sähköpostikansioon.

Viestikäsittelijän tehtävä on suorittaa monimutkainen viestikäsittely. Se selvittää viestin vastaanottajat, kääntää viestin sähköpostiviestiformaatista natiiviksi MICS-viestiksi, pakkaa ja salaa viestit, sekä käsittelee mahdollisen päästä-päähän kuittauksen [13]. Viestikäsittelijä välittää ohjelmistorajapinnalta vastaanotetun viestin natiivina MICS-viestinä reitin-komponentille, sekä välittää reitittimeltä vastaanotetun natiivin MICS-viestin sähköpostiviestinä ohjelmistorajapinnalle.

Reititinkomponentti on MICS-ytimen rajapinta ulkoisille liitännöille (eng. Interface). Liitäntä voi olla esimerkiksi HF-, tai VHF-taajuusalueella hyväksikäyttävä kriisialueella käytetty verkkotekniikka, tai se voi olla esimerkiksi VPN-verkoon kytketty IP-liitäntä. Reititinkomponentin ja liitäntöjen välillä käytetään Netlink-protokollaa, johon on tehty pieniä laajennuksia [31][12].

Reititinkomponentilla on useita tehtäviä: yhteyksien ylläpitäminen liitännöille, viestinvälitys MICS-solmujen välillä, natiiviviestien pilkkominen rajapinnalle sopivaan kokoon, sekä pilkottujen viestiosien mahdollinen uudelleenlähetytys, vastaanotettujen viestiosien yhdistäminen ja liitäntöjen ilmoittaman naapuritiedon kerääminen [12]. Reititinkomponentti välittää viestejä MICS-solmujen kesken, lähettää liitännälle viestikäsittelijältä vastaanotetun viestin, sekä mahdollisesti välittää viestikäsittelijälle toiselta solmulta vastaan-

tetun viestin.

MICS-järjestelmässä ydinkomponenttien välinen kommunikointi on toteutettu D-Bus-rajapintaa [22] hyväksikäyttäen. Valvoja-komponentin tehtävä on tarjota D-Bus-rajapinta komponenttien väliseen viestintään, sekä tarkkailla ydinprosessien tilaa [13].

Reititysydin-komponentin tehtävä on laskea reititys MICS-solmujen välille, sekä välittää reititystietoa MICS-solmujen kesken. Reititysydin toimii IS-IS-reititysprotokollan kaltaisesti ja se mahdollistaa reititystiedon päivittymisen verkkotopologian muuttuessa.

3.2 Reititys

MICS-järjestelmässä osoitteistus on toteutettu kahden tyyppisillä osoitteilla: käyttäjä- ja solmuosoitteilla. Kun viestinkäsittelijä-komponentti vastaanottaa viestin ohjelmistorajapinnalta, se selvittää millä MICS-solmulla viestin vastaanottaja, eli kohdekäyttäjä, sijaitsee. Seuraavaksi viesti annetaan reititinkomponentille, joka reitittää viestin edelleen solmuosoitteen perusteella. Käyttäjien osoitetietoa ylläpidetään LDAP-tietokannassa. Kohdekäyttäjän, sekä kohdesolmun tunnus asetetaan viestinkäsittelijässä natiiviin MICS-viestiin.

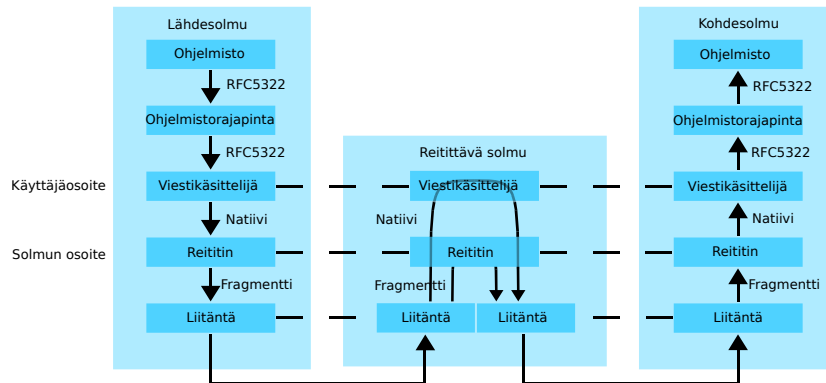
Reititin-komponentti selvittää reititystaulusta parhaan reitin kohdesolmulle. Jos reittiä kohdesolmulle ei ole tiedossa, viesti säilötään siltä varalta, että reititys kohdesolmulle onnistuu myöhemmin. Jos reitti löytyy, reititin-komponentti fragmentoi viestin ja välittää sen liitännälle edelleenlähetettäväksi.

Kuvassa 3.3 esitetään eri ohjelmistokomponenttien roolit MICS-reitityksessä. Lähdesolmulla ohjelmisto välittää viestin ohjelmistorajapinnalle edelleenreititettäväksi. Lähetetty viesti on standardissa sähköpostiformaatissa, joka on määritetty tarkemmin Internet standardissa RFC 5322 [26].

Viestikäsittelijä vastaanottaa viestin ja selvittää viestin vastaanottajan kohdesolmun LDAP-tietokannasta. Viestikäsittelijä muuttaa viestin natiiviksi MICS-viestiksi, enkryptaa ja pakkaa viestin, sekä välittää sen reititettäväksi reititin-komponentille.

Reititin tarkistaa reititystaulusta parhaan reitin kohdesolmulle. Reititin fragmentoi viestin fragmentteihin, jotta viesti voidaan välittää tehokkaasti liitännän yli seuraavalle MICS-solmulle.

Reitittävällä solmulla liitäntä vastaanottaa viestin ja välittää sen reitittimelle. Reititin tarkistaa parhaan reitin kohdesolmulle ja reitittää viestin edel-



Kuva 3.3: MICS-järjestelmän komponentit [13].

leen. Jos viestin reititys vaatii tarkistuksen, viesti ohjataan viestikäsittelijälle prosessoitavaksi. Viestikäsittelijä ohjaa viestin edelleen reitittimelle, kun tarkistus on suoritettu.

Kohdesolmulla viesti välitetään komponentilta toiselle, niin että viesti päättyy asiakasohjelmistolle. Jos viestiin pyydettiin kuittaus, viestinkäsittelijä-komponentti lähettää kuittauksen lähettäjä-solmulle.

Dynaaminen reititys

Dynaaminen reititys MICS-solmujen välillä saavutetaan reititysydin-komponentin avulla. Reititysydin seuraa reitittimen liitäntöjen naapureiden tilaa ja mainostaa naapuritietoa muille MICS-solmuille. Naapuruustieto välitetään muille solmuille MICS-sanomina.

Naapuritiedon levitessä verkon solmuille, jokainen solmu saa tietoonsa verkon rakenteen, eli topologian. Reititysydin selvittää verkon topologiatiedon avulla lyhyimmät reitit muille verkon MICS-solmuille. IS-IS-protokollan tavoin myös MICS-järjestelmän dynaamisessa reitityksessä verkko on jaettu hierarkkisiin kerroksiin topologiaviestien määrän rajoittamiseksi.

3.3 Monilähetys

MICS-järjestelmässä ei itsessään ole monilähetystoiminnallisuutta. MICS-järjestelmän on tarkoitus olla tehokas viestinvälitysjärjestelmä erityisvälitysverkoissa, joissa verkon tarjoama kaista saattaa olla pieni ja yhteydet katko- naisia. Kuten useissa protokollissa aikaisemmin, myös MICS-järjestelmässä on keskitytty ensiluokkaisen ratkaisun kehittämiseen täsmälähetetyksessä, en-

nen kuin tartutaan monilähetyksen ongelmiin.

Monilähetykselle on kuitenkin yleinen tarve viestijärjestelmissä, joita on tarkoitettu käyttämään esimerkiksi kriisialueiden verkoissa. Kriisialueen verkossa viestit ovat usein tarkoitettu useammalle kuin yhdelle vastaanottajalle.

Yksi esimerkki monilähetyksen tarpeellisuudesta nähdään kuvassa 3.1. Tilanteen johto on jaettu kahteen johtokeskukseen: tilannejohtoon ja pääjohtokeskukseen. Kriisialueen käyttäjien tilanteen johdolle välittämät viestit tulee usein lähettää sekä tilanne-, että pääjohtokeskukseen, eli lähetetyillä viesteillä on usein ainakin nämä kaksi vastaanottajaa. Lisäksi johtokeskuksen kriisialueelle välittämät viestit koskevat usein ryhmää alaisia, eivätkä vain yksittäisiä alaisia. Monilähetys on siis tarpeellinen palvelu, koska lähetykset niin alaisilta johdolle, kuin johdolta alaisille tapahtuvat usein useammalle vastaanottajalle.

Käytettävyyshyötyjen lisäksi monilähetys mahdollistaa tehokkaamman resurssien käytön verkossa, jos viestillä on useampi vastaanottaja. Tehokas resurssien käyttö voi olla merkittävää erityisesti niissä verkon osissa, joissa resurssit ovat vähäisemmät. MICS-verkossa resursseja on vähemmän erityisesti verkon laidalla. Verkon laidalla monilähetys mahdollistaa liitännöille erityisten piirteiden hyväksikäytön monilähetyksessä: esimerkiksi radiotiedon siirrossa viesti voidaan välittää usealle vastaanottajalle yhdellä lähetyksellä erillisten lähetysten sijaan.

Ratkaisuvaihtoehdot

Monilähetys on siis tarpeellinen palvelu erityisvälitysverkoissa ja siten myös MICS-järjestelmässä. Monilähetys voidaan toteuttaa MICS-järjestelmään usealla eri tavalla. Eri toteutustavoilla on omat vahvuutensa ja heikkoutensa.

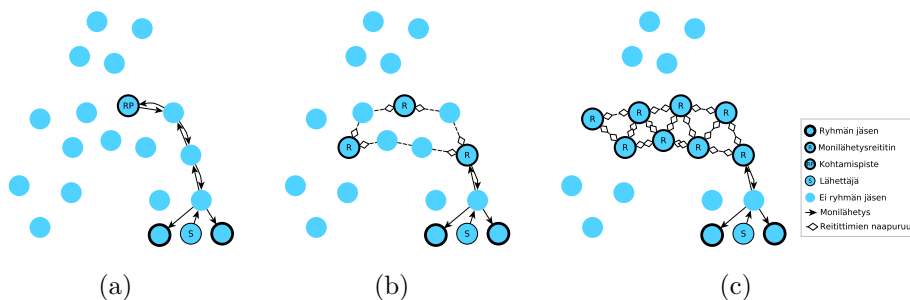
Monilähetys voidaan toteuttaa MICS-järjestelmään lisäämällä monilähetystoiminnallisuus osaksi jonkin, tai joidenkin, ydinkomponenttien toimintaa. Vaihtoehtoisesti monilähetys voidaan toteuttaa MICS-järjestelmän päällä ajettavana sovelluksena.

Lisäksi kaikkien verkon solmujen ei tarvitse tukea monilähetystä: monilähetys voidaan toteuttaa niin, että vain yksi, tai osa, verkon solmuista tukee monilähetystä ja monilähetykset reititetään monilähetykseen kykenevien solmujen kautta.

Keskitetyn solmun ratkaisu

Sovelluksena toteutettuna monilähetys voidaan toteuttaa keskitettynä palveluna yksittäisellä solmulla jolta käyttäjät tilaavat haluamansa ryhmät. Esimerkiksi Internet-verkossa yleiset sähköpostilistat toimivat tällä periaatteel-

la. Lisäksi monilähetyksellä PIM-SM, joka esitettiin luvussa 2.1, käyttää keskitettyä solmua, kohtaamispistettä (eng. Rendezvous Point), jonka kautta monilähetykset välitetään. Kuvassa 3.4(a) esitetään esimerkki kohtaamispisteen käytöstä.



Kuva 3.4: Vaihtoehtoja monilähetyksen toteutukselle

Kohtaamispisteen käyttäminen voi kuitenkin olla epäedullinen valinta haasteellisissa verkoissa. Monilähetyksen toteutus yhden keskitetyn solmun kautta aiheuttaisi monilähetykseen kriittisen pisteen (eng. single-point-of-failure), jonka toimimattomuus estäisi monilähetyksen toiminnan koko verkossa. Lisäksi monilähetyksien tulisi välittyä kohtaamispisteen kautta, minkä vuoksi monilähetykspaketit voidaan joutua välittämään pitkiä etäisyyksiä turhaan, jos lähettäjä-solmu on lähellä vastaanottajaa ja keskitetty solmu kaukana sekä lähettäjästä, että vastaanottajasta.

Kuvan 3.4(a) esimerkissä monilähetyksen kiertää kohtaamispisteen kautta, vaikka lähetyksen olisi voitu reitittää lyhyempääkin reittiä. Kuvan esimerkissä kohtaamispisteen toimimattomuus estäisi monilähetyksen toiminnan. Monilähetyksen reititys epäonnistuu myös, jos joku kohtaamispisteelle reitittävä solmu ruuhkautuu tai lakkaa toimimasta, eikä vaihtoehtoista reittiä ole.

Osa verkon solmuista monilähetyksen palvelimina

Vaihtoehtoisesti monilähetyksen voidaan toteuttaa sovelluksena, niin että monilähetysovellusta ajetaan kaikilla tai osalla MICS-solmuista. Jos monilähetysovellusta ajetaan useammalla kuin yhdellä solmulla, ei monilähetyksessä tarvitse muodostua keskitettyä pistettä, jonka toiminta on edellytys monilähetyksen toiminnalle. Sen sijaan monilähetyksen voidaan toteuttaa hajautetusti niin, että mikä tahansa yksittäinen piste voi lakata toimimasta ilman, että monilähetyksjärjestelmä lakkaa toimimasta.

Kuvassa 3.4(b) on esimerkki monilähetyksen toteutuksesta niin, että osa verkon solmuista on valittu monilähetyksen reitittimiksi. Koska monilähetyksen reititti-

miä on useita, ei yksittäisen monilähetysreittimen toimimattomuus estä monilähetyksen toimintaa koko verkossa.

Kohtaamispisteen käyttöön verrattuna ratkaisu on kuitenkin monimutkaisempi: tulee ratkaista kuinka monilähetysreitittimet valitaan verkossa ja kuinka monilähetysreitittimet välittävät monilähetysviestejä keskenään.

Kuvan esimerkissä monilähetysreitittimet on valittu niin, että ne yhdistävät eri osia verkosta. Kuvan oikean alalaidan monilähetyspalvelin kykenee palvelemaan sekä lähettäjä, että vastaanottajia, koska nämä sijaitsevat samassa verkon osassa. Kuitenkin, jos verkon solmut liikkuvat, saatetaan monilähetykset joutua reitittämään pitempiä matkoja verkossa, jotta ne saadaan reititettyä valitulle monilähetysreitittimelle. Lisäksi, reunasolmujen tulee tietää, mitkä verkon solmut kykenevät monilähetykseen, jotta ne osaavat tarvittaessa vaihtaa valittua monilähetysreititintään.

Monilähetysreitittimien tulee lisäksi välittää tarvittavat monilähetyspaketit toisilleen, jotta monilähetyspaketit saadaan reititettyä verkon osien välillä. Yksi ratkaisu on signaloida reitittimien välillä, mille monilähetysryhmälle milläkin monilähetysreitittimellä on jäseniä ja reitittää vain tarvitut ryhmät. Monilähetysprotokolla PIM-SM-protokolla toimii tällä periaatteella: siinä monilähetysreititin tilaa itselleen ne monilähetysryhmät, joille reitittimellä on tilaajia.

Vaihtoehtoisesti monilähetysreitittimet voivat lähettää toisille monilähetysreitittimille kaikki vastaanotetut monilähetysryhmät. Tällaista monilähetystä kutsutaan tulvittavaksi monilähetykseksi. Yksi tunnettu tulvittava monilähetysprotokolla on PIM-DM. PIM-DM-protokollassa monilähetysreitittimen vastaanotettua monilähetyksen, johon sillä ei ole jäseniä, peruu se jäsenyytensä kyseiseen ryhmään lähettäjältä. Tätä lähetyksen ja perumisen kiertoa kutsutaan tulvita ja karsi -kierroksi (eng. flood and prune cycle), joka esiteltiin tarkemmin luvussa 2.1. Tulvittavassa monilähetyksessä viestien kierto voidaan estää esimerkiksi viestitunnisteita käyttämällä, tai paluupolkuvälityksellä.

Kaikki runkoverkon solmut monilähetysreitittiminä

Kuvassa 3.4(c) esitetään vaihtoehto, jossa kaikki verkon keskusalueen solmut kykenevät monilähetysreititykseen. Kuvan 3.4(b) verrattuna tämän vaihtoehdon hyötynä on, ettei lehtisolmun tarvitse tietää mitkä keskusalueen solmut tukevat monilähetystä, vaan riittää, että lehtisolmu voi käyttää mitä tahansa verkon keskusalueen solmua, eli runkosolmua, monilähetykseen. MICS-järjestelmässä solmut ovat tyypillisesti jakautuneet runko- ja lehtisolmuiksi [12].

3.4 Toteutus - Ydinjärjestelmän komponenttina vai sovelluksena

Monilähetystä toteutettaessa tulee siis valita millä verkon solmuilla monilähetystä tuetaan. Lisäksi tulee valita miten monilähetys toteutetaan MICS-järjestelmään. Monilähetys voidaan toteuttaa joko sovelluksena, joka käyttää MICS-järjestelmän sovelluksille tarkoitettuja IMAP- ja SMTP-rajapintoja. Vaihtoehtoisesti monilähetys voidaan toteuttaa osaksi MICS-järjestelmän ydinkomponentteja tai omaksi ydinkomponenttikseen, joka kommunikoi muiden ydinkomponenttien kesken käyttäen järjestelmän sisäistä D-Bus-rajapintaa.

Toteutus sovelluksena

Sovelluksena toteutettu monilähetys voi käyttää IMAP- ja SMTP-rajapintoja, jotka mahdollistavat viestien lähettämisen ja vastaanottamisen. IMAP- ja SMTP-rajapinnat eivät kuitenkaan tarjoa tietoa järjestelmän sisäisestä tilasta. Ydinkomponenteilla on käytettävissään järjestelmän sisäinen rajapinta D-Bus, jonka välityksellä voidaan tarkkailla, sekä ohjata, muiden ydinkomponenttien tilaa. D-Bus rajapinnan vuoksi MICS-järjestelmän ydinkomponenteilla on käytettävissään tarkempi tieto järjestelmän sisäisestä tilasta.

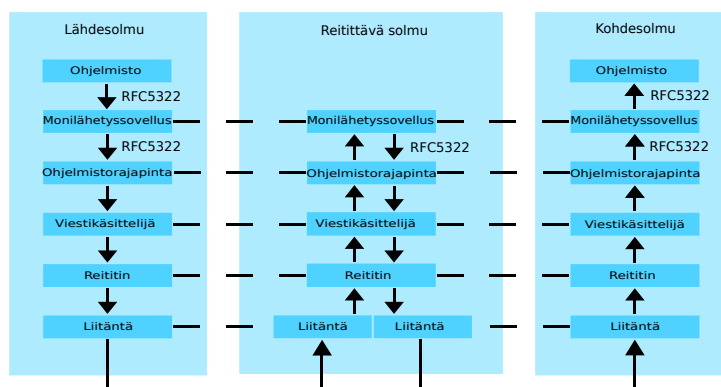
MICS-järjestelmän reititin- ja reititysydinkomponenteilla on käytettävissään reititystietoa, joka mahdollistaa täsmälähetyksien reitittämisen solmujen välillä. Tätä reititystietoa voitaisiin hyödyntää monilähetysreitityksessä. Sovelluksilla on kuitenkin käytettävissään vain IMAP- ja SMTP-rajapinnat viestien lähettämiseen ja vastaanottamiseen. Nämä rajapinnat eivät näytä järjestelmän sisäistä reititystietoa sovelluksille.

Koska ohjelmistona toteutettu monilähetys ei voi hyödyntää MICS-järjestelmän sisäistä reititystietoa, tulee sovelluksen toteuttaa monilähetysreititys erikseen MICS-verkon päälle. MICS-viestivälitysverkon päälle toteutettavaan monilähetysverkkoon on toteutettava joitakin viestinvälitysverkolta vaadittuja ominaisuuksia, kuten kierronesto, erikseen, vaikka alla olevassa MICS-verkossa olisikin kierronesto toteutettuna sisäisesti. Monilähetyssovellus hyötyy kuitenkin MICS-järjestelmän viestisalauksesta ja pakkauksesta.

Monilähetysjärjestelmä voi myös hyväksikäyttää MICS-järjestelmää viestien lähetyksissä. Viestien lähetys monilähetyssoelmujen välillä voidaan toteuttaa yksinkertaisina täsmälähetyksinä MICS-järjestelmää käyttäen. Tällöin monilähetykseen kykenevien solmujen väliset täsmälähetykset hyötyvät MICS-järjestelmän täsmälähetysreitityksestä.

MICS-järjestelmässä tuki verkon dynaamisille muutoksille on toteutettu reititysydin-komponentin avulla. Koska monilähetyssovellus ei käytä järjestelmän sisäistä D-Bus-rajapintaa, ei monilähetyssovellus voi vaikuttaa reititysydin-komponentin toimintaan. Monilähetyssovellus ei siis voi hyödyntää MICS-järjestelmän dynaamista reititystä, vaan sen on toteutettava tuki dynaamisille verkon muutoksille erikseen.

Sovelluksena toteutettu monilähetysjärjestelmä lisää yhden kerroksen MICS-reititykseen. Kuva 3.5 esittää monilähetyssovelluksen monilähetysreitityksen. Täsmäreititykseen verrattuna viestit tulee välittää yhdelle lisäkerrokselle, monilähetyssovellukselle, joka suorittaa monilähetysreitityksen.



Kuva 3.5: Monilähetysreititys sovellus-kerroksen toteutuksessa

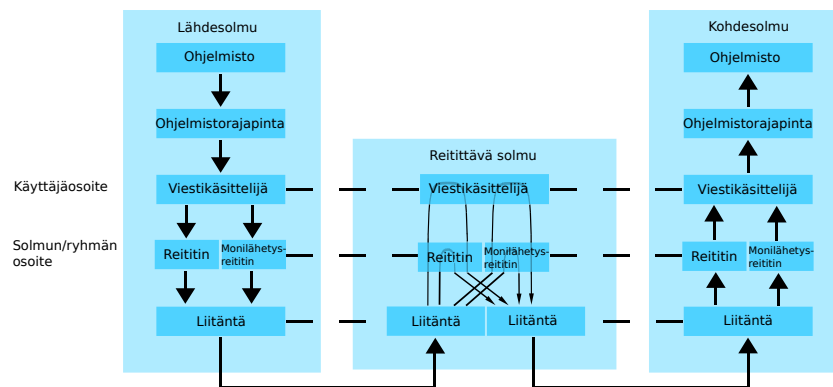
Monilähetysviesti on välitettävä monilähetyssovellukselle myös reitittävällä solmulla, jotta monilähetysreititys voidaan suorittaa. Täsmälähetysreitityksessä, joka esitettiin kuvassa 3.3, viesti purettiin reitittävällä solmulla joko reititin-komponentille, tai varmennuksen vuoksi viestikäsittelijäkomponentille asti. Monilähetyssovelluksessa välitettävä monilähetysviesti on purettava monilähetyssovellukselle asti, eli kaikkien täsmälähetysreititykseen osallistuvien MICS-komponenttien yli.

Viestin välittäminen sovelluskomponentille asti aiheuttaa täsmälähetyskseen verrattuna enemmän kuormaa reitittävällä solmulla. Kuorma on suurempi, koska viesti tulee purkaa natiiviviestistä sähköpostiviestiksi, jotta se voidaan välittää monilähetyssovellukselle. Monilähetyssovellukselta eteenpäin välitettävä monilähetysviesti tulee pakata edelleen natiiviksi viestiksi.

Prosessoinnista aiheutuu lisäkuormaa välittäville solmuille. Riippuu verkon välitysnopeudesta ja välittävien solmujen prosessointinopeudesta, onko lisääntyneestä prosessoinnista haittaa viestinvälityksessä.

Toteutus ydinjärjestelmän komponenttina

Vaihtoehtoisesti monilähetys voidaan toteuttaa osana MICS-järjestelmää, joko omana komponenttinaan, tai osana olemassa olevia komponentteja. Oma komponenttinaan toteutettuna monilähetyksen komponentti voitaisiin toteuttaa reititin-komponentin rinnalla monilähetyksreititystä suorittavana komponenttina. Jos monilähetys toteutettaisiin osana olemassa olevia komponentteja, tulee harkita mihin komponentteihin muutokset on parasta sijoittaa.



Kuva 3.6: Monilähetyksreititys erillisenä ydinkomponenttina

Kuvassa 3.6 esitetään monilähetyksen toteutus erillisenä komponenttina. Täsmälähetykseen verrattuna viestinkäsittelijä- sekä liitäntäkomponenttien tulee tukea kahta reitittävää komponenttia: monilähetyksreititin-, sekä täsmälähetyksreititinkomponenttia. MICS-järjestelmässä tuetaan tällä hetkellä vain yhtä reitityskomponenttia. Muutos saattaa vaatia järjestelmän sisäisten rajapintojen muokkausta.

MICS-järjestelmä sallii myös kolmannen osapuolen toteuttamat liitäntäkomponentit [12]. Näiden kolmannen osapuolen toteuttamien komponenttien muokkaaminen niin, että ne tukevat kahta reitityskomponenttia vain yhden sijaan saattaa olla ylitsepääsemättömän vaikeaa.

Jos monilähetys toteutetaan olemassa olevien ydinkomponenttien sisään, ei liitäntä- ja viestinkäsittelijäkomponenttia tarvitse muokata tukemaan useaa reititinkomponenttia. Tulee kuitenkin harkita mihin komponentteihin monilähetyksreititys on parasta toteuttaa.

Parhaimman komponentin valintaan vaikuttaa merkittävästi osoitteistuksen valinta. MICS-järjestelmä tarjoaa kahden tyyppisiä kohdeosoitteita: käyttäjä- ja solmuosoitteita. Jos monilähetys toteutetaan käyttäjäosoitteistuksella, luontevinta olisi toteuttaa monilähetys osaksi viestinkäsittelijää, joka suorittaa

käyttäjäosoitteiden reitittämisen. Jos monilähetys toteutetaan solmuosoitteita käyttäen, olisi reititinkomponentti luontevin komponentti toteutukselle, koska se on vastuussa solmuosoitteiden reitityksestä. Tulee siis valita kumpaa osoitteitusta käyttää monilähetykselle: käyttäjä-, vai solmuosoitteita.

Monilähetysoikeuden kannalta ryhmän jäsenet ovat aina käyttäjäosoitteita. Vaikka osoitteistukseen käytettäisiin solmuosoitteita, kohdesolmuilla joudutaan selvittämään mille käyttäjälle viesti on tarkoitettu, jotta viesti voidaan toimittaa kohdekäyttäjän sähköpostilaatikkoon. Tämän vuoksi voi olla luontevaa käyttää ryhmäosoitteistuksessa käyttäjäosoitteita.

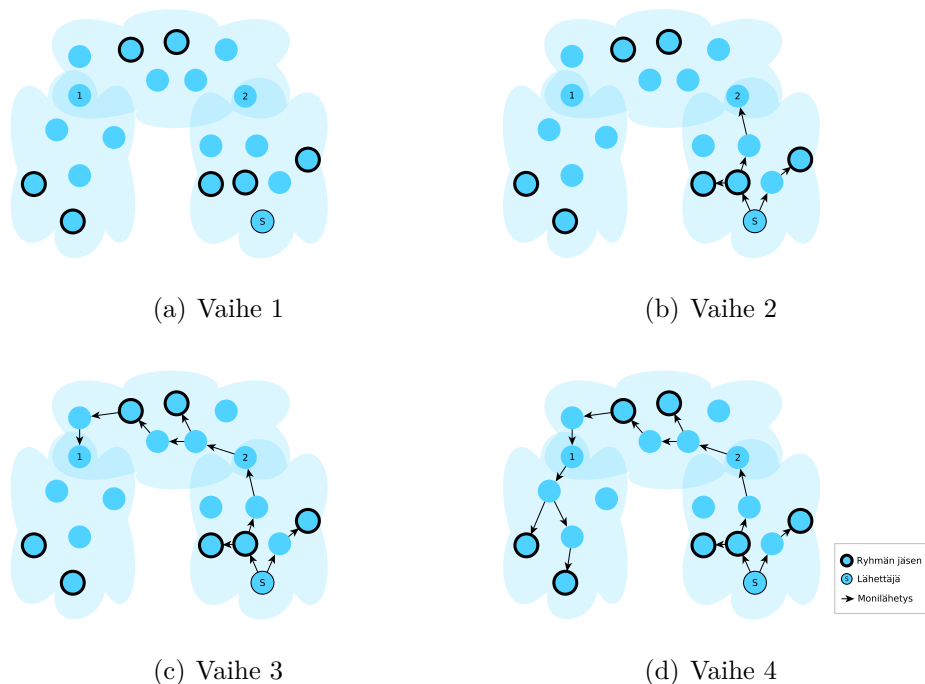
Käyttäjäosoitteita käyttävä monilähetysoikeus vaatii kuitenkin tuen käyttäjätietojen dynaamiseen muuttamiseen. Käyttäjätietoja tulee kyetä muuttamaan dynaamisesti niin, että muutokset käyttäjätiedoissa välittyvät verkon yli muille solmuille. Näin muutokset ryhmäjäsenyyssiedoissa saataisiin välitettyä verkon muille solmuille.

Käyttäjäosoitteiden käyttäminen ryhmien osoitteistamisessa luo kuitenkin haasteen: jos ryhmien jäsenet asetetaan vastaanottajiksi lähetettäviin viesteihin, isoilla ryhmillä viestien vastaanottajien lista kasvaa suureksi. Viestin kohteiden lista saattaa tällöin kasvattaa viestin kokoa merkittävästi. Tämä ongelma voidaan kiertää hyväksikäyttämällä reititysydin-komponentin käyttämää IS-IS reititysoikeusprotokollaa seuraavalla tavalla.

Reititysydin-komponentti jakaa IS-IS-protokollan tavoin verkon hierarkisiin alueisiin. Kuvassa 3.7 on esitetty verkon jako reititysoikeusalueisiin. Verkon jakoa alueisiin voidaan hyödyntää monilähetysoikeudessa lähettämällä monilähetysoikeusviestit yhdelle alueelle kerrallaan.

Lähetäjäsolmun lähettäessä monilähetysoikeusviestin, se merkitsee monilähetysoikeuden vastaanottajaksi kaikki oman IS-IS-alueen solmut, sekä muihin IS-IS-alueisiin reitittävät solmut. Kuvan 3.7(b) esimerkissä lähettävä solmu S lähettää viestin oman alueensa ryhmän jäsenille, sekä ylempään IS-IS-alueeseen reitittävälle solmulle 2.

Ottaessaan vastaan toisen reititysoikeusalueen viestin reititettäväksi, solmu tarkistaa topologiatiedostaan, mitä reittiä reititysoikeushierarkiassa alempi solmu reitittäisi viestin muille tuntemilleen ryhmän jäsenille. Koska viestin vastaanottanut solmu on reititysoikeusallisesti ylemmällä tasolla, se tuntee alemman alueen verkkotopologian. Tämän tiedon avulla viestin vastaanottanut solmu reitittää viestin oman alueen solmuille, sekä solmuille, joiden kautta pääsee alemmille reititysoikeusalueille, joille viestiä ei ole vielä reititetty. Kuvan 3.7(c) esimerkissä solmu 2 lähettää viestin edelleen oman alueensa ryhmän jäsenille, sekä toiselle alemmalle IS-IS-reititysoikeusalueelle reitittävälle solmulle 1.



Kuva 3.7: IS-IS-reititysprotokollan hierarkian hyväksikäyttö monilähetyksreitityksessä. Pilvet esittävät IS-IS-reititysprotokollan alueita.

Reitittäessään itseään IS-IS-reitityshierarkiassa alemmalle tasolle, solmu tarkistaa mitä reittiä viestin lähettänyt solmu olisi reitittänyt solmun ja lähettää viestin edelleen alemman tason ryhmän jäsenille, joille viestiä ei ole vielä reititetty. Lähettäessään viestin solmu merkitsee itsensä viestin lähettäjäksi ja asettaa viestin alkuperäisen lähettäjän viestiin erilliseen kenttään. Solmu asettaa itsensä lähettäjäksi, koska reitityshierarkisesti alempana olevat solmut eivät tunne alkuperäistä lähettäjää, joka on toiselta saman tason reititysalueelta. Kuvan 3.7(d) esimerkissä solmu 1 merkitsee itsensä viestin lähettäjäksi ja tallettaa alkuperäisen lähettäjän viestiin, sekä lähettää viestin oman alueensa ryhmän jäsenille.

Edellä kuvattu menetelmä voitaisiin toteuttaa MICS-järjestelmään muokkaamalla viestinkäsittelijä-komponenttia. Sen tulisi kyetä asettamaan viestin vastaanottajat edellä kuvatun suunnitelman mukaisesti. Ryhmien jäsentieto olisi kuvattuna viestikäsittelijän käyttämässä LDAP-tietokannassa.

Kuvattu menetelmä kuitenkin vaatii mekanismin ryhmien jäsenyystietojen välittämiseen verkon muille solmuille. Ilman ryhmäjäsenyystiedon välittymistä verkossa, ryhmien jäsenyystiedot eivät voi muuttua verkon olemassa-

lon aikana. Jos MICS-järjestelmään toteutetaan menetelmä käyttäjätietojen välittämiseen verkossa, voi monilähetyksen toteuttaminen edellä kuvatulla tavalla olla järkevää. Ilman olemassa olevaa mekanismia jäsenyystietojen välittämiseen, voi yksinkertaisen monilähetysjärjestelmän toteuttaminen erillisenä sovelluksena MICS-järjestelmän päälle olla yksinkertaisempi vaihtoehto.

3.5 Yhteenveto

Monilähetysjärjestelmän toteuttaminen MICS-järjestelmän päälle käyttää enemmän järjestelmän resursseja, kuin järjestelmän sisäinen toteutus, koska verkossa välitettävät viestit tulee purkaa monilähetyssovellukselle jokaisella monilähetystä reitittävällä solmulla. Lisäksi sovelluksena toteutettava monilähetysjärjestelmä ei voi hyväksikäyttää MICS-järjestelmän sisäistä reititystietoa monilähetysreitityksen optimoimiseksi.

Sovelluksena toteutettu monilähetysjärjestelmä ei kuitenkaan aiheuta muutoksia MICS-järjestelmän sisäiseen toimintaan ja se pystytään toteuttamaan täysin erillisenä komponenttina. Koska MICS-järjestelmässä ei ole tukea käyttäjäosoitteiden muutosten välittämiseksi verkon yli, voi olla yksinkertaisempaa toteuttaa monilähetysjärjestelmä MICS-verkon päälle.

Tässä työssä suunniteltu ja toteutettu monilähetysjärjestelmä toimii sovelluksena MICS-järjestelmää käyttäen. Se rakentaa yksinkertaisen monilähetysreitityskerroksen MICS-verkon solmujen päälle. Runkoverkon solmujen kesken monilähetysjärjestelmä yksinkertaisesti tulvittaa viestit solmujen välillä. Verkon reunalla viestit välitetään vain ryhmän tilanneille solmuille.

Luku 4

Toteutus ja validointi

Tässä luvussa esitellään erityisvälitysverkkoihin suunniteltu monilähetysjärjestelmä nimeltä TMR (eng. Tactical Message Replicator). TMR-järjestelmän käyttöliittymä perustuu sähköpostilistapalvelimien toimintamalliin. Jokaisella TMR-solmulla on oma jakelulista, jonka jäsenille vastaanotetut viestit välitetään. Erikoisuutena sähköpostilistapalvelimiin verrattuna TMR-jakelulistan jäsenenä voi olla myös toisen solmun TMR-sovellus. Näin ollen eri solmuilla ajettavat TMR-sovellukset kykenevät välittämään monilähetysviestit keskenään kohti lopullisia monilähetysryhmän jäseniä.

Viestien kiertäminen, niiden välittyessä TMR-solmujen välillä, on estetty usealla kierronestomekanismilla. Monilähetysryhmään listaudutaan lähettämällä paikalliselle TMR-sovellukselle ryhmätilausviesti, jolloin paikallinen TMR-sovellus lisää lähettäjän jakeluryhmän jäseneksi. Lisäksi TMR-sovellus suorittaa tarvittavat operaatiot verkon suuntaan, jotta viestien välitys sekä verkosta käyttäjälle, että käyttäjältä verkkoon onnistuu.

4.1 TMR-verkon toiminta

Kaikkia TMR-monilähetysryhmiä ei välttämättä haluta välittää kaikkien TMR:n kesken. Jos kaikki monilähetysryhmät välitettäisiin kaikkien TMR-solmujen kesken, viestit toimitettaisiin kaikille TMR-solmuille, eli viestit tulvitettaisiin verkkoon. Tämä saattaa olla toivottu toimintamalli, jos halutaan optimoida viestien perille päätyminen todennäköisyys verkon resurssienkäytön kustannuksella.

TMR-järjestelmässä verkon oletetaan koostuvan kahdesta osasta: runkoverkosta ja siihen liittyvistä liikkuvista solmuista. Runkoverkossa oletetaan ole-

van enemmän kapasiteettia viestinvälitykseen, kuin siihen liittyvillä liikkuvilla solmuilla. Liikkuvat solmut liittyvät runkoverkkoon esimerkiksi VHF-radioliitännällä, runkoverkon muodostuessa suuremman siirtokaistan yhteyksistä kuten mikroaaltolinkeistä ja kuituyhteyksistä. Koska verkon laidalla on vähemmän kaistaa viestinvälitystä varten, TMR-järjestelmän kaistankäytön tulee olla konservatiivisempaa liikkuvilla solmuilla kuin runkoverkossa.

Runkoverkossa kaikki monilähetysviestit välitetään kaikkien muiden runkosolmujen kesken. Tällöin runkoverkon tehtäväksi jää välittää monilähetysryhmien viestit runkoon liittyville liikkuville tilaajille. Välitettäessä yksinkertaisesti monilähetysviestit kaikkien runkosolmujen kesken, vältetään monimutkaisen monilähetysprotokollan toteuttamiselta runkoverkossa.

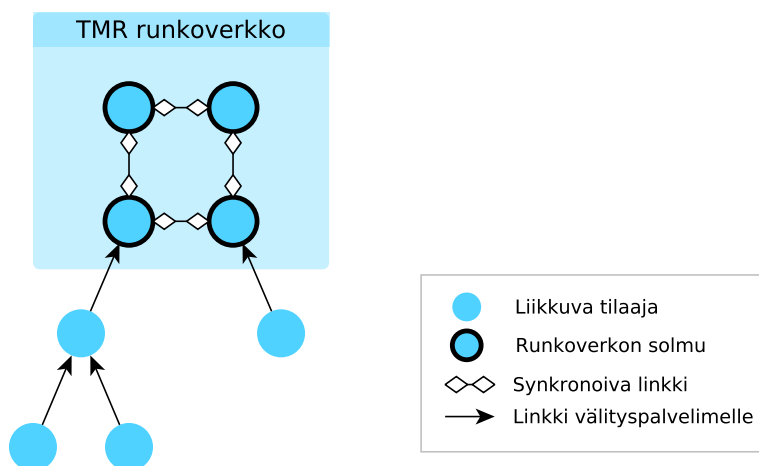
Koska liikkuvien tilaajien käytettävissä olevan siirtokapasiteetin oletetaan olevan rajallinen, kaikkia runkoverkon monilähetysryhmiä ei kannata lähettää liikkuville solmuille. Liikkuville solmuille tulee lähettää vain ne monilähetysryhmät, joita joku kyseisen liikkuvan solmun käyttäjä tilaa. Jotta käyttäjän tilaamat ryhmät kyetään edelleentilaamaan joltain runkoverkon solmulta, tarvitaan keino ilmoittaa TMR-sovellukselle miltä solmulta edelleentilata ryhmätilaukset. TMR-välityspalvelin tarjoaa tämän toiminnallisuuden.

TMR-välityspalvelin

TMR-välityspalvelimella ilmoitetaan miltä naapurisolmulta TMR-sovelluksen tulisi edelleentilata uudet ryhmätilaukset. TMR-välityspalvelimet on tyyppillisesti asetettu niin, että ne muodostavat ketjun runkoverkkoon liittyviltä solmuilta runkoverkkoon päin. Esimerkitapaus on esitetty kuvassa 4.1. Runkoverkossa TMR-sovellukset lähettävät kaikki monilähetysviestit toisilleen. Liikkuvien tilaajien TMR-välityspalvelimet on asetettu niin, että välityspalvelimet muodostavat ketjun runkoverkkoon.

Vastaanottaessaan ryhmätilauksen, TMR-sovellus edelleentilaa ryhmän TMR-välityspalvelimeltaan. Esimerkki ryhmätilauksesta on esitetty kuvassa 4.2. Ryhmätilauksen jälkeen solmulle välitetään ryhmän viestit rungosta. Vastaanottaessaan viestin, liikkuvan tilaajan TMR-sovellus välittää viestin TMR-välityspalvelimilleen, joten kaikki liikkuvien tilaajien lähettämät viestit välittyvät runkoon.

Tilauksen jälkeen ryhmään lähetetyt viestit lähetetään välityspalvelimelle, koska kaikki liikkuvan solmun monilähetysviestit lähetetään TMR-välityspalvelimelle. Lisäksi TMR-välityspalvelimen monilähetysryhmään vastaanotetut viestit lähetetään liikkuvan solmun TMR-sovellukselle, koska liikkuvan solmun TMR-sovellus on edelleentilannut kyseisen ryhmän TMR-välityspalvelimeltaan.



Kuva 4.1: Esimerkki TMR-verkosta. Synkronoitavalla linkillä kaikki monilähetysviestit lähetetään solmujen kesken.

Jos liikkuva tilaaja halutaan irrottaa TMR-välityspalvelimesta ja siirtää toisen TMR-välityspalvelimen liikkuvaan tilaajaksi, tulee irrotettavan solmun signaloida irtautuminen vanhalle TMR-välityspalvelimelleen. TMR-välityspalvelimen tulee saada tieto siitä, ettei irrotettavalle solmulle tule enää lähettää ryhmäviestejä.

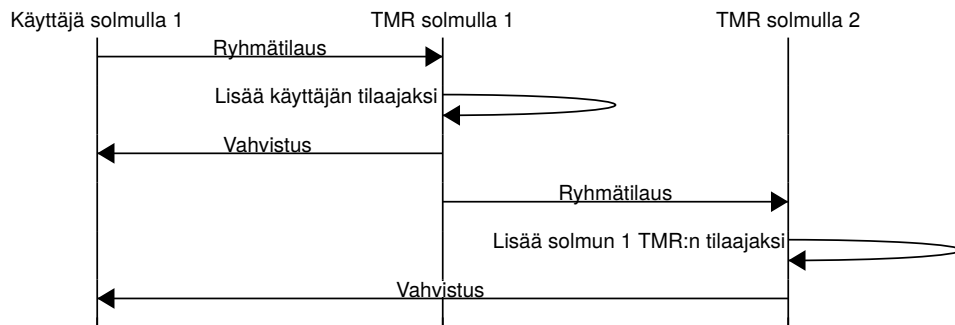
Kierronesto

Kierronesto on TMR:ssä toteutettu kolmella mekanismilla: paluupolkutiedolla, viestitunnisteella (eng. Message-ID, MID) ja TTL-kentällä (eng. Time-To-Live).

Paluupolkutieto

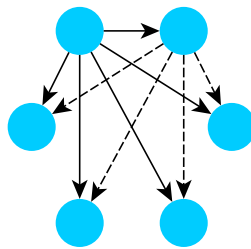
Paluupolkutiedolla TMR:n välittämiin viesteihin lisätään tieto millä solmuilla viesti on jo käynyt. Jos vastaanotetusta viestistä nähdään, että viesti on jo käynyt solmulla, viesti hylätään. Lisäksi viestiä välitettäessä, ei viestiä lähetetä paluupolun solmuille.

Jos TMR-verkko on konfiguroitu täysin kytketyksi verkoksi, eli niin että kaikki verkon solmut näkevät toisensa, voi verkossa muodostua ylimääräisiä lähetyksiä. Tämä on seurausta siitä, että täysin kytketyssä verkossa viestin vastaanottaessaan solmu välittää viestin kaikille solmuille joiden kautta viesti ei ole vielä kulkenut. Tilanne on kuvattu kuvassa 4.3. Viesti siis välitetään myös solmuille joille se on lähetetty edellisen lähetyksen yhteydessä. Ongelmaa voidaan lieventää sisällyttämällä lähetettäviin viesteihin tieto siitä, mille



Kuva 4.2: Esimerkki TMR ryhmätilauksesta. Solmun 1 TMR edelleentilaa tilausryhmän välityspalvelimeltaan solmulla 2. Lisäksi solmun 1 TMR lisää solmun 2 TMR:n ryhmän tilaajaksi.

solmuille viesti on aikaisemmin lähetetty, paluupolkutiedon lisäksi.



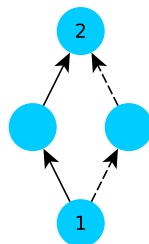
Kuva 4.3: Esimerkki ylimääräisistä lähetyksistä käytettäessä paluupolkukierroa. Ensimmäisen lähetyksen (merkitty jatkuvalla viivalla) vastaanottaja edelleenlähettää (merkitty katkoviivalla) viestin kaikille muille paitsi lähettäjälle, vaikka viesti on jo lähetetty kaikille solmuille.

Ylimääräisten lähetyksien välttämiseksi viesteihin kootaan aikaisempien lähetyksien lista. Aikaisempien lähetyksien lista koostetaan solmuista joille viesti on jo lähetetty. Listalta voidaan kuitenkin poistaa ne solmut, jotka löytyvät paluupolkulistalta. Tämä on seurausta siitä, että viesti on varmasti lähetetty jokaiselle solmulle joka sen on vastaanottanut, jolloin sama tieto tulisi muuten talletettua kahdesti.

Viestitunniste

Paluupolkukierro on riittävä kierronestomekanismi, jos viesti ei voisi välittyä solmujen välillä kuin yhtä reittiä. Jos viesti voi välittyä useampaa kuin yhtä reittiä saatetaan sama viesti käsitellä useamman kerran. Kuvassa

4.4 esitetään tilanne jossa sama viesti käsitellään kahdesti. Viesti välitetään ensin kahdelle solmulle, jotka välittävät sen edelleen yhdelle solmulle. Tämä solmu käsittelee kyseisen viestin kahdesti, koska katkoviivalla esitettyä reittiä saapunut viesti ei sisällä jatkuvalla nuolella merkityllä reitillä välittyvästä viestistä ja päinvastoin.



Kuva 4.4: Esimerkki paluupolkukierrosta useamman polun tapauksessa.

Viestitunnisteen käyttö estää edellä mainitun skenaarion. Viestitunnisteita käytettäessä viestien kierto estetään ylläpitämällä listaa välitettyjen viestien viestitunnisteista ja hylkäämällä vastaanotetut viestit joiden viestitunniste löytyy listalta. Kuvan 4.4 skenaario ei ole viestitunnisteita käytettäessä mahdollinen, koska ensimmäisenä solmulle 2 saapuvan viestin viestitunniste talletetaan ja toista reittiä pitkin saapuvaa viestiä ei käsitellä, koska sen viestitunniste löytyy käsiteltyjen viestien listalta.

Time-To-Live

TMR-järjestelmässä TTL-kenttää vähennetään yhdellä jokaisella TMR-solmulla ja sen laskiessa nollaan viesti hylätään. Jos viesti kiertää verkossa solmulta toiselle, laskee TTL-kentän arvo lopulta nollaan, jolloin viestiä ei käsitellä ja kierto loppuu. TTL-kentän arvo voidaan asettaa monilähetysryhmäkohtaisesti, tai käyttää järjestelmän oletusarvoa.

TTL-kenttä on hyödyllinen lisäominaisuus myös viestinvälityksen osalta, jos esimerkiksi halutaan välittää viesti vain lähistöllä sijaitseville solmuille. Tällöin TTL-kentälle voidaan antaa pieni arvo ja viesti ei leviä koko verkkoon vaan vain lähellä sijaitseville solmuille.

Verkon rakenteen optimointi

Kuvan 4.1 esimerkissä runkoverkko on konfiguroitu täysin kytketyksi verkoksi. Runkoverkkoa ei ole välttämätöntä konfiguroida näin, vaan se voidaan konfiguroida tarpeen mukaan esimerkiksi tähdeksi tai kehäksi. Valitussa konfiguraatiossa tulee ottaa huomioon alemmalla kerroksella käytetty verkkotek-

niikka ja mahdolliset topologiamuutokset.

4.2 Toiminnan testaus ja arviointi

MICS-järjestelmän päällä ajettavan monilähetysjärjestelmän oikea toiminta riippuu usean komponentin toiminnasta. Kaikkien reititykseen osallistuvien MICS-järjestelmän komponenttien on toimittava odotetulla tavalla, jotta monilähetys toimisi oikein. Lisäksi itse TMR-järjestelmän toiminta on testattava.

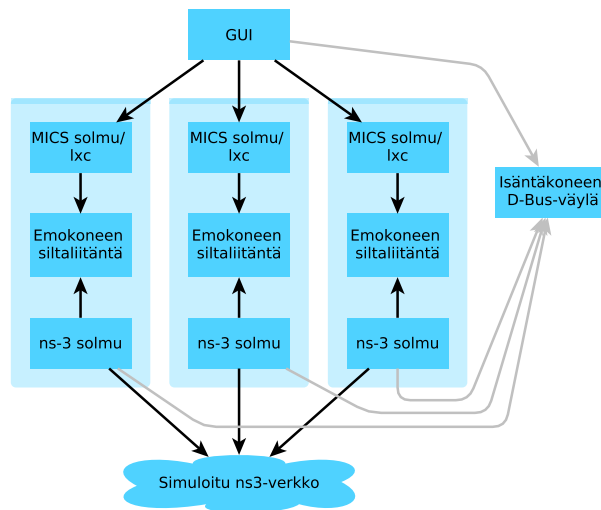
Koska TMR:n toiminta riippuu myös alla ajettavan järjestelmän oikeasta toiminnasta, on järkevää testata halutut monilähetyskennäriiot kokonaisuutena. Testauksessa otetaan käyttöön kokonainen testiverkko, mukaan lukien testisolmut ja testisolmujen väliset fyysiset yhteydet. Lopuksi arvioidaan järjestelmän suoriutuminen testistä.

Testaustyökalut

Monilähetysten testaukseen toteutettiin erillinen testausohjelmisto. Testausohjelmistolla on tavoitteena simuloida solmuja, sekä solmujen välisiä yhteyksiä mahdollisimman todenmukaisesti. Koska todellisessa kriisitilanteen verkossa solmut tyypillisesti liikkuvat, tulee myös testiverkon solmujen kyetä liikkumaan. Näin voidaan testata monilähetysten toiminta myös yhteyksien katketessa ja syntyessä uudestaan verkossa. Mahdollisuus erotella runko- ja lehtisolmut on myös tärkeää, jotta voidaan mallintaa runko- ja lehtisolmuihin jakautunutta MICS-verkkoa todenmukaisesti.

Kuvassa 4.5 kuvataan testijärjestelmän arkkitehtuuri. Testausohjelmisto koostuu suurelta osin valmiista ohjelmistoista. Solmujen liitäntöjenvälistä yhteyttä mallinnetaan yleisesti käytetyllä ns-3 verkkosimulointiohjelmistolla [21]. Ns-3 mahdollistaa verkon reaaliaikaisen simuloinnin. Lisäksi ns-3 mahdollistaa testiohjelman toteuttamisen Python skriptikielellä [9], joka mahdollistaa testiohjelmistojen toteuttamisen nopeallakin aikataululla. Ns-3 simulaattorissa on valmiit toteutukset useille fyysisen ja MAC-kerroksen verkkoprotokollille.

Itse MICS-testisolmut simuloidaan lxc-virtuaalikoneiden avulla [1]. Perinteisiin virtualisointialustoihin verrattuna lxc-virtualisointi mahdollistaa huomattavasti kevyemmän resurssienkäytön. Perinteisissä virtualisointiratkaisuissa tyypillisesti virtualisoidaan koko käyttöjärjestelmän alainen järjestelmä niin, että kokonaista käyttöjärjestelmää ajetaan virtualisoidun tietokoneen päällä. Lxc-virtualisoinnissa hyväksikäytetään Linux-käyttöjärjestel-



Kuva 4.5: Monilähetksen testausohjelmiston arkkitehtuuri kolmella testisolmulla solmulla

män ominaisuuksia niin, että virtualisoitua järjestelmää ei tarvitse ajaa virtualisoidun tietokoneen päällä. Sen sijaan virtualisoitu järjestelmä ajetaan osana isäntäkoneen käyttöjärjestelmää niin, että se näkee itsellään isäntäkoneesta erilliset resurssit.

Testijärjestelmässä jokaiselle simuloitavan MICS-verkon solmulle tulee asettaa omat IP-osoitteet. Solmuille tarvitaan oma IP-osoite hallintaa varten ja yksi jokaista simuloitavaa IP-verkkoyhteyttä varten. Tässä testissä kaikki yhteysvälit käyttävät IP-yhteyttä.

Lisäksi jokaisella MICS-solmulla tulee olla virtualisoituna hakemisto ajonaikaisen tiedon tallettamiseen, jotta järjestelmä voi tallettaa prosessoitavat viestit tiedostojärjestelmälle. Jokaiselle virtualisoidulle MICS-solmulle varataan isäntäkoneelta kansio ajonaikaisen tietojen tallettamista varten. Ajonaikaisia tietoja varten varattu kansio näyttäytyy lxc-virtuaalikoneen sisällä MICS-järjestelmän normaalina ajonaikaisten tietojen kansiona, vaikka isäntäkoneella se sijaitsee solmulle erityisesti varatussa sijainnissa.

Lxc-virtuaalikoneen ja ns-3 verkkosimulaattorin sisäisesti mallinnettu solmu kytketään toisiinsa isäntäkoneen siltaliitännän (eng. bridge-interface) avulla. Siltaliitäntä välittää Ethernet-paketit siltaan kytkettyjen liitännöiden välillä. Lxc-virtuaalikone kytkee itsensä siltaliitännään virtuaalisen liitännän kautta. Ns-3-solmun liitännää varten luodaan oma virtuaalinen liitäntä, niin kut-

suttu tap-liitäntä (eng. tap-interface), joka liitetään siltaliitäntään ja johon ns-3-solmu kytketään. Näin MICS-solmun lxc-virtuaalikoneen sisällä lähetämä Ethernet-paketti välittyy siltaliitännän kautta ns-3-solmulle ja edelleen simuloitavaan ns-3-verkkoon.

Lxc-virtuaalikoneessa virtualisoidaan jokaiselle solmulle oma hallintaverkko-liitäntä, verkkoliitännät langattomaan kommunikointiin lehtisolmujen välille, sekä erillinen liitäntä runkosolmujen väliseen kommunikointiin. Hallintaliitännän kautta kyetään isäntäkoneen kautta viestimään virtualisoidun solmun kanssa. Solmujen keskenäinen kommunikointi hallintaliitännän kautta on estetty isäntäkoneessa.

Lehtisolmujen välinen kommunikointi välittyy simuloitun ns-3-verkon kautta. Solmujen välinen pakettien välittyminen ns-3-verkossa riippuu solmun sijainnista, sekä käytetystä verkkotekniikasta. Runkosolmut kytketään tämän testin testiverkossa ns-3-verkon kautta virtuaalisella lankaliitännällä toisiinsa. Langallisen liitännän vuoksi runkosolmut kykenevät saavuttamaan toisensa riippumatta niiden sijainnista verkossa.

Ns-3-verkkosimulaattorissa ajettava testiohjelma mainostaa solmujen sijainnit isäntäkoneen D-Bus-väylälle. Näin isäntäkoneessa ajettava graafinen käyttöliittymä kykenee lukemaan solmujen sijainnit ja esittämään ne käyttäjälle. Käyttöliittymä ottaa yhteyttä lxc-virtualisoituihin solmuihin hallintaliitännän kautta ja tarkistaa MICS-solmujen tilat suorittamalla kyselyjä MICS-järjestelmän monitori-komponentin D-Bus-rajapinnalle.

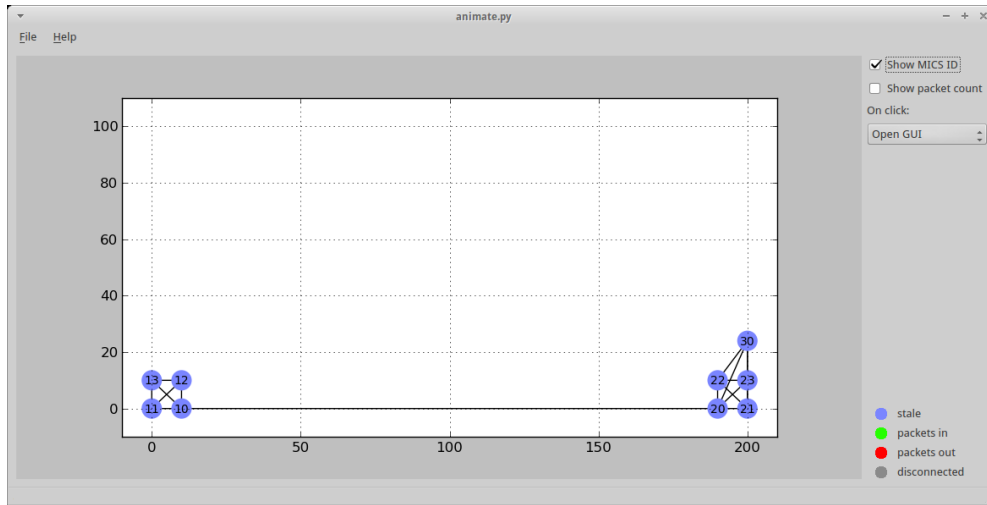
Kuvassa 4.6 esitetään testausohjelmiston käyttöliittymä. Kuvaajan akseleilla käytetty yksikkö on metri. Käyttöliittymässä esitetty verkko on jakautunut kahteen osaan: vasempaan ja oikeanpuoleiseen verkkoon. Vasemmalla esitetyn verkon jäsenet ovat solmut 10, 11, 12 ja 13. Oikean puolen verkon jäseniä ovat solmut 20, 21, 22, 23 ja 30. Solmujen välillä on viiva, jos niillä on toisilleen naapuruus, eli ne ovat joko toistensa langattoman yhteyden kantoalueella, tai niillä on toisiinsa langallinen yhteys.

Alueiden sisäiseen liikennöintiin solmut käyttävät langatonta WLAN-liitäntää, joka toimii IEEE standardin 802.11a mukaisesti. Alueiden väliseen liikenteeseen käytetään virtuaalista langallista liitäntää. Alueet ovat toisiensa WLAN-liitäntöjen kantoalueen ulkopuolella, mutta ne kykenevät kommunikoidaan runkosolmujen langallisen liitännän välityksellä. Alueiden sisällä solmut kykenevät kaikki kommunikoidaan keskenään langattomasti.

Testi 1: Monilähetyksen toiminta ja kierronesto

Motivaatio

TMR-monilähetyksjärjestelmän tarjoama komentorajapinta noudattelee tyy-



Kuva 4.6: Monilähetyksen testausohjelmiston käyttöliittymä

pillistä sähköpostilistasovelluksen rajapintaa. Käyttäjä voi tilata itselleen monilähetyksryhmän lähettämällä tilausviestin TMR-sovellukselle, joka kuittaa tilauksen käyttäjälle. TMR-solmujen välittäessä monilähetyksiä verkossa, kierroneston avulla estetään sanomien kiertäminen, jos verkossa on silmukoita.

Tämän testin tarkoitus on testata, että TMR-monilähetyksjärjestelmä reagoi odotetulla tavalla käyttäjän tilausviesteihin ja että järjestelmä välittää monilähetyksiä onnistuneesti tilaajille. Testissä varmistetaan lisäksi, että kierronesto toimii odotetusti.

TMR-monilähetyksjärjestelmässä kierronesto on toteutettu kolmea mekanismia hyväksikäyttäen: viestitunnisteet, aikaisempien lähetysten lista, sekä paluupolkutieto. Kierroneston toteuttaminen näiden kolmen mekanismin avulla on suoraviivaista: solmujen tulee ylläpitää listaa jo käsitellyistä viestitunnisteista, sekä tarkistaa viestiä välitettäessä, ettei viestiä lähetetä solmuille, joille se on lähetetty aiemmin.

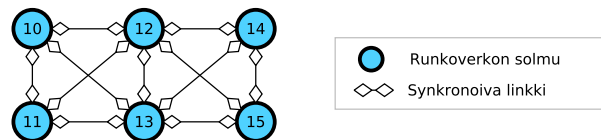
Vaihtoehtoinen ratkaisu olisi rakentaa monilähetykspuu lähetyksiä varten. Monilähetykspuu voi olla joko jaettu, tai lähdesolmukohtainen. Reititettäessä lähdesolmukohtaisen monilähetykspuun mukaisesti viestit välitetään kohdesolmuille lyhyintä reittiä pitkin. Monilähetykspuun rakentaminen vaatii kuitenkin solmujen välistä signalointia, sekä naapuruustiedon välittämistä, jotta solmut saavat riittävän tiedon verkon rakenteesta monilähetykspuun rakentamiseksi.

Lähdekohtaiset monilähetyspuut ovat optimaalinen kierronestoratkaisu, koska niitä käytettäessä kohdesolmut saavutetaan lyhyintä polkua käyttäen. TMR-monilähetysjärjestelmä käyttää kuitenkin yksinkertaisempaa kierronestoratkaisua toteutuksen yksinkertaistamiseksi, sekä kierronestoon tarvittavan signaloinnin vähentämiseksi.

Koska TMR-monilähetysjärjestelmässä ei käytetä optimaalista monilähetyspuuta reitityksessä, tulee testata kuinka paljon verkkoon aiheutuu ylimääräisiä lähetyksiä valitun kierronestoratkaisun vuoksi.

Toteutus

Testi suoritetaan kuvan 4.7 mukaisella topologialla. Testin kaikki solmut ovat runkosolmuja, eli ne välittävät kaikki vastaanottamansa monilähetysviestit toisilleen. Ensimmäisessä vaiheessa lähetetään tilausviesti jokaiselta solmulta erikseen ja varmistetaan, että käyttäjä saa tilauksesta kuittausviestin. Kun tilauksen onnistuminen on varmistettu, lähetetään monilähetysryhmään 100 testiviestiä kullakin solmulla ja varmistetaan, että muut solmut vastaanottavat lähetetyt testiviestit.



Kuva 4.7: TMR-järjestelmän topologia ensimmäisessä testissä. Kaikki solmut ovat runkosolmuja.

Solmujen lähettämien viestien määrä tarkistetaan ennen ja jälkeen 100 viestin lähettämisen. Kierronestossa käytettävien mekanismien perusteella voidaan antaa arvio, mikä lähetettyjen viestien määrä tulisi olla jokaisella solmulla. Täsmällisen lähetysmäärän ennustaminen on mahdotonta, koska viestit välittyvät verkossa toisistaan riippumatta ja ei voida ennakoida missä järjestyksessä viestit saapuvat millekin solmulle. Arvion perusteella voidaan kuitenkin arvioida toimiiko kierronesto odotetusti.

Lisäksi lasketaan kuinka monta lähetystä olisi tehty, jos TMR reitittäisi lähdekohtaisen monilähetyspuun mukaisesti. Näin voidaan verrata TMR:n kierronestomekanismia optimaaliseen kierronestoratkaisuun.

Optimaalisessa ratkaisussa, lähdekohtaisessa levityspuussa lähetyksiä tehtäisiin vähemmän kuin TMR-monilähetyksessä. Lähdekohtaisessa levityspuussa laitasolmujen 10 ja 11, sekä 14 ja 15, monilähetykset edelleenlähetettäisiin

vain kerran yhtä vastakkaisen laidan solmua kohden. TMR-monilähetyksessä runkoverkon viestien reititys perustuu tulvitukseen, jolloin molemmat verkon keskusalueella sijaitsevat solmut, 12 ja 13, edelleenlähettäisivät monilähetysviestin molemmille vastakkaisen laidan laitasolmuille. Laskelman mukaan lähdekohtaisessa levityspuussa lähetyksiä muodostuu 5 lähetystä yhtä monilähetysviestiä kohden, TMR-monilähetyksen tehdessä 7 lähetystä yhtä laitasolmun monilähetystä kohden, sekä 5 lähetystä yhtä keskusalueen monilähetystä kohden. Tulvitusmekanismin vuoksi odotetussa tuloksessa TMR-järjestelmä suorittaa siis optimaaliseen ratkaisuun verrattuna kaksi ylimääräistä lähetystä jokaista laitasolmun lähettämää monilähetysviestiä kohden.

Lopuksi varmistetaan tilauksen perumisen toiminta. Jokaisen solmun tilaaja lähettää tilauksen perumisviestin solmunsä TMR-sovellukselle, minkä jälkeen tarkistetaan kuittausviestin saapuminen. Lisäksi ryhmään lähetetään viesti ja tarkistetaan, että lähetetty viesti välittyy runkosolmujen välillä oikein ja ettei lähetetty testiviesti saavu käyttäjille.

Tämän testin perusteella voidaan arvioida toimiiko TMR:n komentorajapinta, monilähetysviestien välittyminen ja kierronesto oikein. Testin tulosten perusteella voidaan arvioida kykeneekö TMR-järjestelmä toimittamaan viestit tilaajille onnistuneesti ja toimiiko kierronesto odotetusti.

Testi 2: Edelleentilaus, sekä tukiaseman vaihto reunasolmulla

Motivaatio

TMR-monilähetyksjärjestelmässä viestien välittyminen liikkuvalla solmulla toteutetaan asettamalle liikkuvalla solmulla tukiasemasolmu. Käyttäjän tilatesä monilähetyksryhmän oman solmunsä TMR-sovellukselta, se edelleentilaä monilähetyksryhmän tukiasemasolmultaan.

Tukiasemia käyttäen monilähetyksviestejä välitetään verkossa pienempi määrä, kuin skenaariossa jossa käyttäjät tilaisivat ryhmän keskitetyltä solmulla verkossa. Viestejä välitetään vähemmän, koska topologisesti tilanneen solmun alla sijaitsevien solmujen ei tarvitse tilata ryhmää keskitetyltä solmulla, vaan ryhmän tilannut solmu voi välittää ryhmän alaspäin. Lisäksi, jos solmulla on useampi sisäinen käyttäjä, ei samaa viestiä tarvitse reitittää verkossa jokaiselle solmun käyttäjälle erikseen, vaan riittää, että viesti reititetään solmulla kerran.

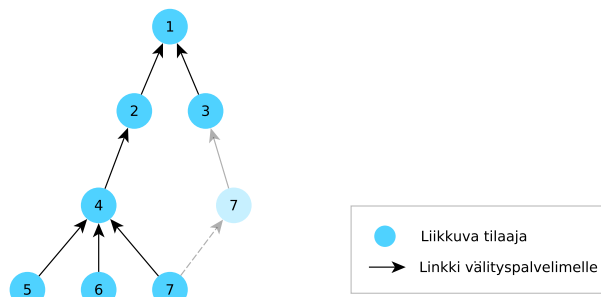
Monilähetyksen toteuttaminen tukiaseman avulla aiheuttaa kuitenkin lisäongelman: tukiasema tulee kyetä asettamaan ajonaikaisesti ja tukiasema tulee kyetä vaihtamaan siten, että aikaisemmat tilaukset siirretään uudelle tukiasemalle.

Tässä testissä varmistetaan, että TMR-palvelinsovellus kykenee onnistunees-

ti vaihtamaan tukiasemasolmua. Testissä varmistetaan, että solmun aiemmin tilaamat monilähetysryhmät tulevat tilattua uudelta tukiasemasolmulta ja että tilaus poistetaan vanhalta tukiasemasolmulta. Lisäksi näytetään tukiasemia hyödyntävän ratkaisun hyöty keskitetyn solmun monilähetysratkaisuun verrattuna.

Toteutus

Testin topologia on esitetty kuvassa 4.8. Testin alussa liikkuva solmu 7 asettaa tukiasemasolmukseen solmun 4. Tämän jälkeen varmistetaan kuittausviestin saapuminen. Seuraavaksi liikkuva solmu lähettää tilauksen oman solmunsu TMR-sovellukselle, minkä jälkeen tarkistetaan, että varmistusviesti saapuu sekä tukiasemasolmulta, että paikalliselta solmulta. Lisäksi solmulle 3 tilataan ryhmä kahdelle ylimääräiselle solmun sisäiselle käyttäjälle. Tämän jälkeen lähetetään testiviestejä yhdeltä solmulta kerrallaan kaikkiaan 100 kappaletta jokaiselta solmulta ja varmistetaan viestien saapuminen kaikille verkon käyttäjille.



Kuva 4.8: TMR-järjestelmän topologia toisessa testissä. Solmu 7 vaihtaa tukiasemaansa solmusta 4 solmuun 3.

Seuraavaksi verrataan lähetettyjen monilähetysviestien määrää tilanteeseen, jossa käytetään keskitettyä monilähetyssovelmaa, ilman tukiasemia. Kaikki solmut tilaavat monilähetysryhmän suoraan solmulta 1. Ryhmä tilataan myös kahdelle solmun 3 sisäiselle lisäkäyttäjälle. Tämän jälkeen lähetetään testiviestejä jokaiselta solmulta vastaavasti kuin aikaisemmin ja verrataan solmujen lähettämien viestien määrää ratkaisuun, jossa tukiasemat olivat käytössä.

Lopuksi varmistetaan tilauksien siirtyminen uudelle tukiasemalle liikkuvan solmun vaihtaessa paikkaa verkossa. Ensin asetetaan uusi tukiasema liikkuvalle solmulle lähettämällä sille tukiaseman vaihtava komentoviesti ja tarkistetaan. Seuraavaksi tarkistetaan, että tukiaseman vaihdosta on saapunut kuittausviesti.

Liikkuvan solmun TMR-sovellus edelleentilaa tukiaseman vaihdon yhteydessä ryhmät uudelta tukiasemalta, sekä peruu tilaukset edelliseltä tukiasemalta. Seuraavaksi tarkistetaan, että uudelta tukiasemalta saadaan kuittaus tilatusta ryhmästä. Lisäksi varmistetaan, että tilaus on poistunut vanhalta tukiasemalta.

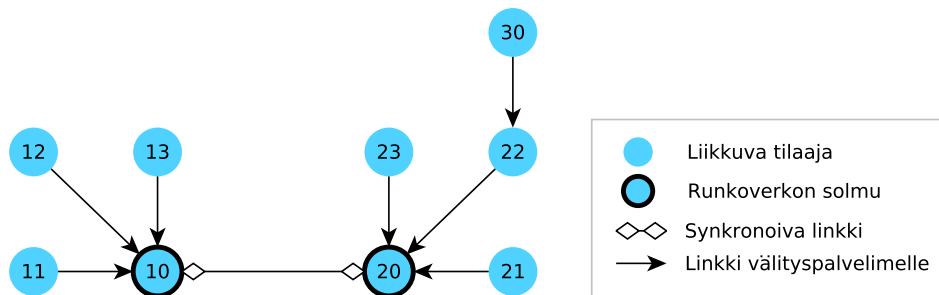
Testin tuloksena voidaan varmistua TMR-järjestelmän oikeasta toiminnasta liikkuvien solmujen osalta. Lisäksi voidaan verrata tukiasemien käyttöön perustuvan monilähetyksen hyöty verrattuna keskitetyn monilähetyssolmun ratkaisuun.

Testi 3: Monilähetysviestien välittyminen katkonaisessa verkossa ja monilähetyssryhmän vanhenemisajan säätö

Motivaatio

MICS-järjestelmä kykenee toimittamaan viestit kohdesolmuille myös katkonaisissa verkoissa. Katkoksien ilmetessä MICS-järjestelmä säilöttää viestiä, kunnes toimiva reitti havaitaan. Kaikkia viestejä ei ole toivottavaa säilöttää loputtomasti, koska viestillä ei välttämättä ole merkitystä enää tietyn ajan kuluttua, tai viestin ei muuten haluta kuluttavan verkon resursseja.

Viestin vanhenemisajan rajoittaminen on tärkeää erityisesti reaaliaikaisissa palveluissa, jotka lähettävät viestejä usein, mutta joiden lähettämällä viesteillä on merkitystä vain rajatun ajan. MICS-järjestelmää käyttävän monilähetysjärjestelmän on tärkeää tukea viestin vanhenemisajan säätöä, jotta voidaan välttää verkon ruuhkautuminen vanhenevista viesteistä verkkokatkoksien aikana.



Kuva 4.9: Testin 3 topogia. Solmu numero 30 liikkuu niin, että se poistuu välillä solmun 22 kantoalueelta.

Toteutus

Tässä testissä varmistetaan monilähetysviestien perille välittyminen katko-

naisissa verkoissa, sekä viestien vanheneminen kun monilähetysryhmän viestien vanhenemisaika on asetettu pieneksi. Testin topologia on esitetty kuvaajassa 4.9. Solmu numero 30 liikkuu kahden pisteen väliä, poistuen välillä tukiasemasolmun 22 kantoalueella.

Testin ensimmäisessä vaiheessa lähetetään jokaiselta solmulta testiviestejä yhdeltä solmulta kerrallaan sekunnin välein niin, että jokaiselta solmulta lähetetään 30 testiviestiä. Viestien vastaanottoajat kirjataan ylös. Ryhmä, johon viestit lähetetään on asetettu käyttämään pitkää vanhenemisaikaa, jotta viestien tulisi välittyä katkoksista huolimatta. Lopuksi varmistetaan, että kaikki viestit ovat saapuvat kaikille verkon solmuille.

Seuraavaksi testataan lyhyemmän vanhenemisajan ryhmäviestien vanheneminen verkossa. Solmut asetetaan lähettämään testiviestejä samoin kuin edellä, mutta tällä kertaa ryhmään jonka vanhenemisaika on asetettu yhteen minuuttiin. Viestien tulisi siis pudota, jos kohdesolmuun ei saada yhteyttä minuutin aikana. Lähetettyjen viestien lähetysajat ja määrät kirjataan ylös, samoin kuin viestien vastaanottoajat.

Testin perusteella voidaan antaa arvio TMR-monilähetysjärjestelmän kyvyttöä säätää monilähetysryhmien vanhenemisaika. Jos vanhenemisaikaa voidaan säätää oikein, voidaan ennaltaehkäistä verkon ylikuormittumista, erityisesti reaaliaikaisten sovellusten lähettämistä viesteistä. Toisaalta vanhenemisajan säädöllä voidaan myös taata tärkeiden ryhmien viestien välittyminen myös katkonaisessa verkossa.

Testi 4: Resurssien käyttö

Motivaatio

Monilähetysjärjestelmää tulee pystyä ajamaan kaikilla MICS-järjestelmän solmuilla. Erityisvälitysverkoissa solmujen käytössä olevat resurssit saattavat olla rajalliset, erityisesti verkon laidoilla. Koska TMR-sovellus jakaa resurssit muun MICS-järjestelmän ja muiden solmulla ajettavien sovellusten kanssa, on tärkeää ettei TMR-sovellus kuluta tarpeettomasti solmun resursseja.

Toteutus

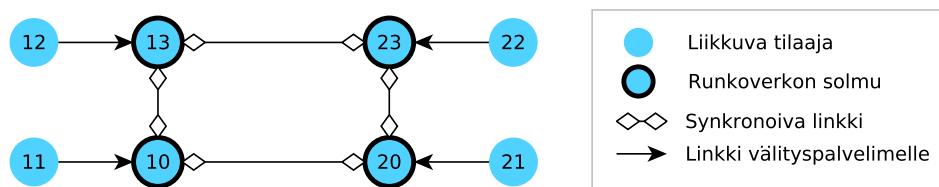
Tässä testissä tarkistetaan TMR-sovelluksen resurssienkäyttö. Resurssienkäyttö tarkistetaan tarkkailemalla sovelluksen muistin-, sekä prosessorin käyttöä testin aikana.

Testissä käytetään kuvan 4.10 topologiaa, jossa on neljä runkosolmua, sekä neljä liikkuvaa solmua. Testiverkko toteutetaan simulaattorilla, lukuunottamatta solmuja 22 ja 23, jotka toteutetaan todellisilla testilaitteistolla. Lehtisolmua 22 ajetaan matalamman suorituskyvyn sulautetulla ALIX (AMD GeodeLX, 800MHz) PC:llä, jossa on asennettuna Debian GNU/Linux

6.0.1 käyttöjärjestelmä. Runkoverkon solmua 23 ajetaan korkeamman suorituskyvyn PC:llä (Intel Xeon W5580, 3,2 GHz), jossa on asennettuna Ubuntu GNU/Linux 13.04 käyttöjärjestelmä.

Ensimmäisessä testissä verkkoa kuormitetaan lähettämällä solmulta 22 kaikkiaan 1000 viestiä maksiminopeudella. Lähetyksen aikana mitataan solmujen 23 ja 22 TMR-sovelluksen prosessorin-, sekä muistinkäyttö.

Toisessa testissä lähetetään testiviestejä yhdeltä solmulta kerrallaan ja tarkistetaan sovelluksen muistinkäyttö testin aikana. Viestejä lähetetään yhdeltä solmulta 5 kappaletta kerrallaan ja testin aikana kaikki solmut kierretään 10 kertaa. Kaikkiaan lähetetään siis 800 viestiä. Testi simuloi tilannetta, jossa jokainen verkon solmu lähettää viestejä nopeassa tahdissa. Testin aikana kirjataan ylös solmujen 22 ja 23 TMR-prosessin muistin- ja prosessorinkäyttö.



Kuva 4.10: Testin 4 topogia. Solmujen muistin-, sekä prosessorinkäyttö testin aikana kirjataan ylös.

Tulokset

Testi 1: Monilähetyksen toiminta ja kierronesto

Testin ensimmäisessä vaiheessa varmistettiin tilauksen toiminta. Testi suoritettiin Python-kielellä toteutetulla testiohjelmalla joka liitti testikäyttäjän MICS-järjestelmään IMAP- ja SMTP-rajapintojen kautta käyttäen Python-kielen standardikirjastoihin kuuluvia smtpplib ja imaplib-kirjastoja.

Tilauksen toiminta testattiin jokaiselta solmulta erikseen lähettämällä testiryhmään tilausviesti ja varmistamalla, että tilaukseen saatiin kuittaus. Jos kuittaus vastaanotettiin onnistuneesti, suoritettiin sama testi seuraavalla solmulla. Jokaisen tilauksen jälkeen odotettiin yksi sekunti kuittausviestin saapumista. Jos kuittausta ei saatu, tilaus todettiin epäonnistuneeksi.

Testissä tilaus onnistui jokaisella solmulla odotetusti, eli jokaiseen kuudesta lähetetystä tilausviestistä vastaanotettiin kuittaus tilauksesta.

Seuraavaksi testattiin kykeneekö TMR-monilähetysjärjestelmä välittämään

Solmu	Vastaanotettu	Lähetetty
10	700	300
11	700	300
12	500	1300
13	500	1300
14	700	300
15	700	300

Taulukko 4.1: Vastaanotettujen ja lähetettyjen viestien määrä TMR-monilähetysjärjestelmän kierronestolla

viestejä onnistuneesti monilähetysryhmän jäsenille. Edellisen testin testisovellusta muokattiin niin, että sovellus lähettää yhden viestin testiryhmään ja varmistaa, että ryhmän jäsenet vastaanottavat viestin. Jos joku ryhmän jäsenistä ei onnistu vastaanottamaan testiviestiä 90 sekunnin kuluttua lähetyksestä, lähetys todettiin epäonnistuneeksi. Testin luotettavuuden kasvattamiseksi sama testi suoritetaan yhtäjaksoisesti 100 kertaa.

Kaikki lähetetyt testiviestit vastaanotettiin kohdesolmuilla onnistuneesti. Taulukossa 4.1 on esitetty solmujen testin aikana vastaanottamat ja lähettämät viestit. Viestien määrät täsmäävät laskennallisen tuloksen kanssa, eli kierro- nesto toimi testissä odotetusti.

Laskennallinen lähetettyjen ja vastaanotettujen viestien määrä voidaan selvittää seuraavalla tavalla: Jokaiselta solmulta lähetettiin yhteensä 100 testiviestiä. TMR-kierronesto sisällyttää viesteihin tiedon, mille solmuille viesti on jo lähetetty. Kulmasolmujen lähetettyjen viestien määrä on 300 jokaisella, koska solmut toimivat lähettäjinä ainoastaan omille testiviesteilleen. Solmuilla on 3 naapuria ja 100 lähetettävää viestiä, joten lähetystyksiä tulee 300. Kulmasolmut eivät edelleenlähetä mitään naapurisolmultaan saamiaan viestejä, koska kaikki kulmasolmun naapurit ovat myös toistensa naapureita. Näin ollen viesti on jo välitetty kaikille kulmasolmun naapurisolmuille edellisellä hypyllä.

Verkon keskiosan solmujen 12 ja 13 lähetetyksien määrä on 1300, koska ne välittävät edelleen kaikilta kulmasolmuilta vastaanotetut viestit. Jokainen kulmasolmu lähettää kummallekin keskussolmulle 100 viestiä, jotka kaikki välitetään edelleen verkon toisen puolen kulmasolmuille. Vastaanotetusta 100:sta testiviestistä muodostuu siis 200 lähetystä. Kulmasolmuja on 4, eli kaiken kaikkiaan lähetystyksiä muodostuu 800, sekä yhteensä 500 omista lähetyksistä, koska keskussolmuilla on 5 naapuria.

TMR-monilähetysjärjestelmän kierronestomekanismi ei ole optimaalinen kier-

Solmu	Vastaanotettu	Lähetetty
10	500	300
11	500	300
12	500	500
13	500	1300
14	500	300
15	500	300

Taulukko 4.2: Vastaanotettujen ja lähetettyjen viestien määrä lähdekohtaisilla monilähetyspuilla

ronesto lähetettyjen viestien määrässä. Optimaalinen kierronestomekanismi lähetettyjen viestien määrässä olisi lähdekohtaiset monilähetyspuut, koska niitä käytettäessä viestit reititetään kohteille lyhyintä mahdollista reittiä pitkin. Taulukossa 4.2 on esitetty tulokset käytettäessä lähdekohtaisia monilähetyspuita.

Lähdekohtaisia monilähetyspuita rakennettaessa kaikkien linkkien hinta oletettiin samaksi. Näin ollen lähdekohtaiset monilähetyspuut saadaan muodostettua yksinkertaisesti estämällä jompaa kumpaa keskussolmuista 12, tai 13 välittämästä edelleen monilähetyspaketteja. Testissä solmulta 12 estettiin edelleenvälittäminen.

Tuloksista nähdään, että kaikki solmut vastaanottivat optimaalisen määrän viestejä: 100 viestiä jokaiselta muulta verkon solmulta, eli 500 viestiä. Kulmasolmujen lähetysmäärät pysyivät samana. Myös keskussolmun 13 lähetysmäärä pysyi samana, koska se oli kaikkien kulmasolmujen lähettäjäkohtaisessa monilähetyspuussa välittäen kulmasolmujen viestit verkon toisen puolen kulmasolmuille. Solmun 12 lähetetyksien määrä putosi 500:aan, koska se ei enää välittänyt edelleen kulmasolmujen viestejä. Solmu 12 lähetti ainoastaan omat 100 testiviestiään kaikille 5:lle naapurilleen, eli tehden kaikkiaan 500 lähetystä.

Kaikkiaan lähettäjäkohtaisia levityspuita käytettäessä lähetysten määrä oli 800 kappaletta, eli noin 21% pienempi kuin TMR:n kierronestoa käytettäessä. Tulos on riippuvainen topologiasta, mutta yksittäisenä tuloksenakin se antaa aihetta harkita TMR-monilähetysjärjestelmän kierronestomekanismien tehostamista. Lähetettyjen viestien määrä on 21% pienempi tehokkaammalla kierronestolla, mutta toisaalta TMR-monilähetysjärjestelmän kierronesto ei vaadi erillisten signaalintiviestien lähettämistä solmujen välillä, kuten lähettäjäkohtaisia levityspuita muodostaessa tulisi tehdä.

Lopuksi testattiin tilauksen perumisen toiminta. Tilaus peruttiin jokaiselta

solmulta ja ryhmään lähetettiin testiviesti jokaiselta solmulta. Lähetyksen jälkeen odotettiin 10 sekuntia ja varmistettiin, ettei millään solmulla ollut ryhmän sanomia. Testin tulos oli odotusten mukainen: yhdellekään entiselle tilaajalle ei ollut saapunut ryhmään lähetettyjä testiviestejä.

Testi 2: Edelleentilaus, sekä tukiaseman vaihto reunasolmulla

Yksi TMR-monilähetysjärjestelmän keskeisistä ominaisuuksista on kyky asettaa tukiasemasolmu, jolta omat tilaukset tulee edelleentilata. Tukiasemasolmuja käyttäen monilähetysviestit voidaan välittää vastaanottajille pienemmällä määrällä lähetyksiä kuin keskitetyn monilähetyssohmun ratkaisussa. Tässä testissä testataan tukiasemasolmujen käytöstä seuraava hyöty, sekä tuki tukiaseman vaihdolle.

Testi aloitetaan lähettämällä jokaiselta solmulta tilausviesti testiryhmään. Käyttäjä saa tilaukseen kuittauksen paikalliselta TMR-sovellukselta, sekä tukiasemasolmultaan. Myös tukiasemien tulee tilata ryhmä omalta tukiasemasolmultaan ja myös tästä tilauksesta lähetetään kuittausviesti käyttäjälle.

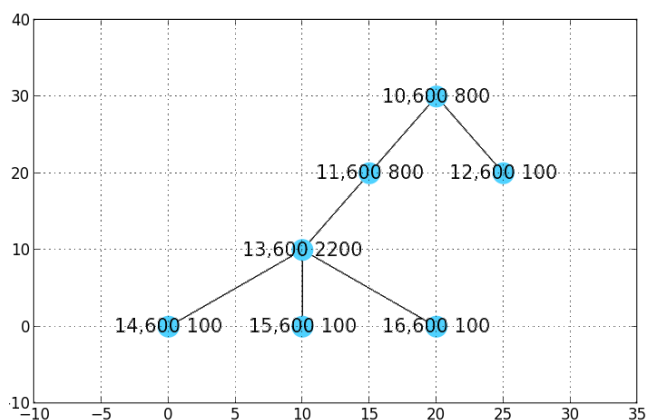
Tukiasemien ketjuttumisen vuoksi solmuille odotetaan eri määrä kuittausviestejä: solmuille 14, 15 ja 50 odotetaan neljää kuittausviestiä, solmulle 13 kolme kuittausviestiä, solmulle 11 ja 12 kaksi kuittausviestiä ja solmulle 10 yksi kuittausviesti. Testissä solmuilta lähetettiin tilausviesti jokaiselta testisolmulta, minkä jälkeen odotettiin 3 sekuntia kuittausviestien saapumista. Jokaiselle solmulle saapui oikea määrä kuittausviestejä.

Tässä testin vaiheessa testissä jokaisella solmulla on ollut yksi testikäyttäjä joka on tilannut testiryhmän. Jotta voidaan tutkia tukiasemaratkaisun toimintaa useamman käyttäjän tapauksessa, lisätään solmulle 12 kaksi käyttäjää ja tilataan testiryhmä myös näille käyttäjille.

Kuvassa 4.11 esitetään testiverkon topologia ja taulukossa 4.3 on listattu vastaanotettujen ja lähetettyjen viestien määrät. Tulokset täsmäävät odotetun tuloksen kanssa.

Lehtisolmuilla 12, 14, 15 ja 16 on lähetettynä 100 viestiä kullakin, jotka muodostuvat tukiasemalle edelleenreititettäväksi lähetetyistä 100:sta omasta testiviestistä. Solmuilla 10 ja 11 on molemmilla 800 lähetystä, jotka muodostuvat seuraavasti: 200 lähetystä omista testiviesteistä (100 molemmille naapureille), sekä 600 viestiä välitettäessä muiden 6 verkon solmujen viestejä lähettäneeltä naapurilta toiselle naapurille.

Solmulla 13 on kaikkiaan 2200 lähetystä, jotka muodostuvat samoin kuin edelliset lähetykset, mutta yhden naapurin lähetykset tulee nyt välittää edelleen kolmelle naapurille, mistä tulee 1800 lähetystä. Lisäksi omat lähetykset tulee lähettää neljälle naapurille, mistä tulee 400 lähetystä, eli kaikkiaan 2200



Solmu	Vast.	Läh.
10	600	800
11	600	800
12	600	100
13	600	2200
14	600	100
15	600	100
16	600	100

Kuva 4.11 & Taulukko 4.3: Vastaanotetut ja lähetetyt viestit käytettäessä tukiasemasolmuja. Kuvan akseleilla etäisyydet metreissä. Solmujen kohdalla ilmoitetaan tiedot järjestyksessä: solmun tunnus, vastaanotetut viestit, lähetetyt viestit.

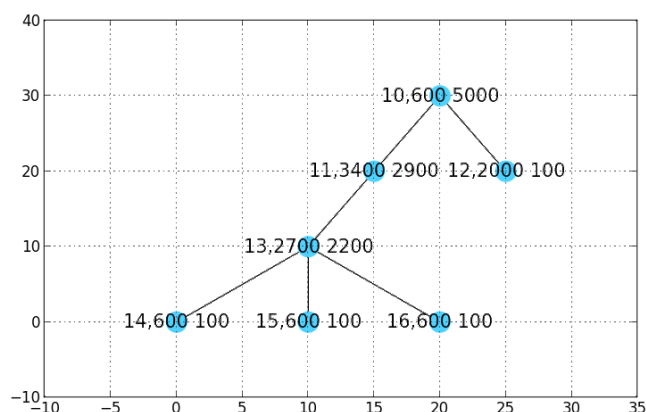
lähetystä.

Jokainen testisolmu vastaanotti 600 viestiä, eli solmut vastaanottivat kerran jokaisen muun solmun lähettämät testiviestit. Tukiasemia hyväksikäyttävän ratkaisun vuoksi solmuille lähetettiin testiviestit vain kerran, vaikka solmulla oli useampi tilaaja. Lisäksi tukiasemana toimivat solmut välittivät testiviestit ainoastaan kerran alla oleville solmuilleen, eikä jokaiselle alla olevalle solmulle lähetetty viestiä erikseen.

Kuvassa 4.12 ja taulukossa 4.4 on esitetty testin tulokset käytettäessä keskitettyä monilähetyssohmua. Tulokset vastaavat ennustettua tulosta ja näytävät, että monilähetysten toteuttaminen tukiasemien avulla vähentää huomattavasti verkossa lähetettyjen viestien määrää, verrattuna keskitetyn monilähetyssohmun ratkaisuun.

Lähetettyjen viestien määrät muodostuvat seuraavasti: lehtisolmut lähettävät kaikki vain omat 100 testiviestiä. Solmu 10 lähettää omat 100 testiviestiä kahdeksalle ryhmäjäsenelle, eli 800 viestiä (solmulla 12 on kolme jäsentä). Lisäksi solmu lähettää muiden testisolmujen lähettämät 600 testiviestiä ryhmän jäsenille, lukuunottamatta lähettäjä, eli seitsemälle jäsenelle, mistä tulee 4200 lähetystä. Kaikkiaan solmulla 10 on siis lähetystiä 5000.

Solmu 11 lähettää ylöspäin omat, sekä allaan olevien solmujen testiviestit, eli 500 testiviestiä. Solmun alla on neljä lähettäjä, joiden testiviestit lähetetään alas muille kuin lähettäjälle, mistä tulee 1200 viestiä. Lisäksi alas lähetetään kaikille neljälle solmulle muiden kolmen solmun testiviestit, eli 1200 viestiä. Kaiken kaikkiaan lähetystiä tulee siis 2900. Solmun 13 lähetystiä



Solmu	Vast.	Läh.
10	600	5000
11	3400	2900
12	2000	100
13	2700	2200
14	600	100
15	600	100
16	600	100

Kuva 4.12 & Taulukko 4.4: Vastaanotetut ja lähetetyt viestit käytettäessä keskitettyä solmua. Kuvan akseleilla etäisyydet metreissä. Solmujen kohdalla ilmoitetaan tiedot järjestyksessä: solmun tunnus, vastaanotetut viestit, lähetetyt viestit.

muodostuvat samalla tavoin ja myös sen lähetyksien määrä täsmää lasketun lähetyksen määrän kanssa.

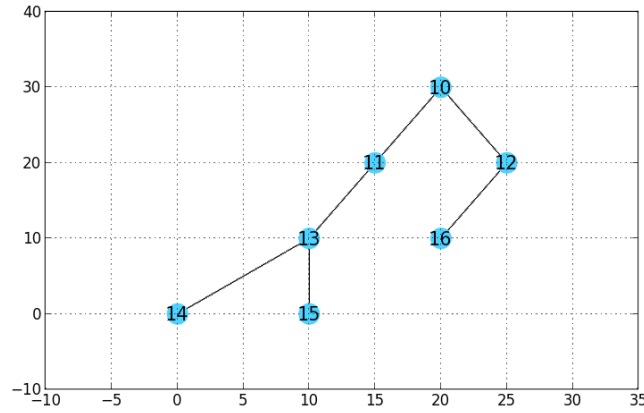
Kaikkiaan ilman tukiasemia verkossa lähetetyksiä oli 10500, kun tukiasemia käytettäessä lähetetyksiä oli 4200. Tukiasemia käytettäessä verkossa oli siis 60% vähemmän lähetyksiä kuin keskitetyn monilähetyssohmun ratkaisussa.

Seuraavaksi testattiin solmun liikkuminen verkossa. Solmu 16 liikkuu verkossa solmun 3 alle. Verkon uusi rakenne on esitetty kuvaajassa 4.13. Solmulle 16 lähetetään tukiasemanvaihtoviesti ja tarkistetaan, että se saa kuittauksen uudelta tukiasemaltaan. Lisäksi tarkistetaan, että vanha tukiasema on poistanut solmun tilaukset ja että uusi tukiasema on muodostanut tilaukset. Kaikki testit onnistuivat odotetusti: solmu sai tukiaseman vaihdon jälkeen kuittausviestin uudelta tukiasemaltaan ja tilaukset olivat poistuneet vanhalta tukiasemalta ja lisäksi uudelle tukiasemalle oli muodostunut tilaukset.

Testi 3: Monilähetysviestien välittyminen katkonaisessa verkossa ja monilähetysryhmän vanhenemisajan säätö

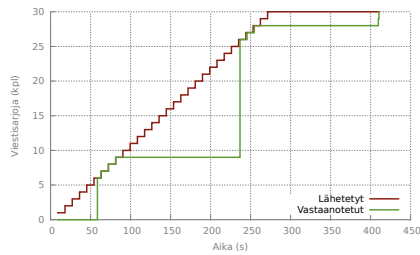
MICS-viestinvälitysjärjestelmässä on mahdollista säätää aikaa, minkä järjestelmä säilöittää välitettäviä viestejä. On tärkeää, että MICS-järjestelmää hyväksikäyttävä monilähetysjärjestelmä hyväksikäyttää viestin säilöittämisen säätöä, jotta voidaan estää verkon ruuhkautuminen vähemmän tärkeistä viesteistä.

Tässä testissä varmistetaan, että TMR-monilähetysjärjestelmä kykenee käyttämään MICS-järjestelmän viestinvanhenemisen säätöä oikein. Testissä käytetään kuvan 4.9 kuvan mukaista verkkoa. Solmu 30 liikkuu kahden pisteen

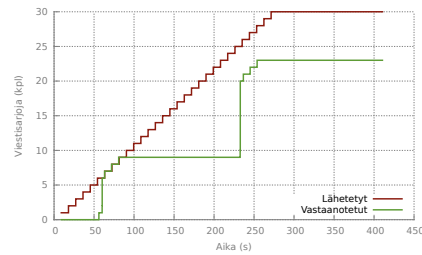


Kuva 4.13: Verkon rakenne solmun 16 liikuttua solmun 12 alle. Kuvan akseleilla etäisyydet metreissä.

välillä poistuen välillä tukiasemasolmunsa 22 kantoalueelta. Etäisyys on pitimmillään 160 ja lyhyimmillään 10 metriä, langattoman liitännän kantoalueen ollessa noin 20 metriä.



(a) Pitkä vanhenemisaika



(b) Lyhyt vanhenemisaika

Kuva 4.14: TMR-monilähetys kahdella vanhenemisajalla

Aluksi lähetetään testiviestejä ryhmään, jolle on määritetty pitkä vanhene-misaika. Tulos on esitetty kuvaajassa 4.14(a). Lähetettyjen viestien määrä kasvaa lineaarisesti, mutta viestejä ei saada toimitettua kaikille vastaanotta-jille solmun 30 poistuessa tukiasemansa kantoalueelta. Viestit eivät kuiten-kaan ehdi vanhentua ja solmun palattua kantoalueelle kaikki viestit saadaan toimitettua.

Kuvassa 4.14(b) esitetään monilähetyksen toiminta ryhmällä, jolle on ase-tettu lyhyt vanhenemisaika. Vanhenemisaika on asetettu yhteen minuuttiin. Nyt kaikkia viestejä ei saada toimitettua vaikka solmu palaa kantoalueel-le. Vanhimmat viestit ovat ehtineet vanhentua, joten MICS-järjestelmä on

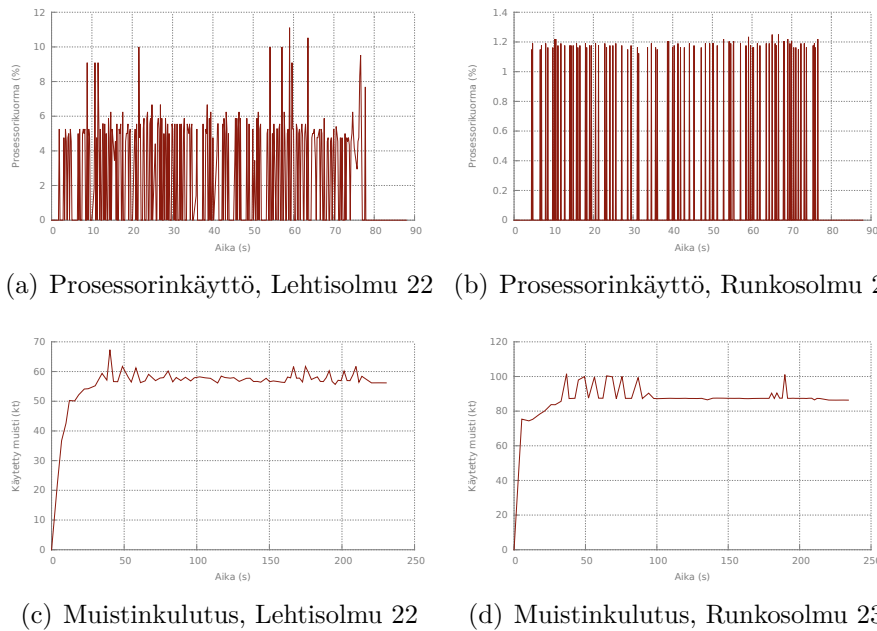
poistanut ne reititysjonosta.

Testin tuloksesta voidaan todeta, että TMR-monilähetysjärjestelmä kykenee hyväksikäyttämään MICS-järjestelmän vanhenemisajan säätöä oikein. Näin monilähetysryhmille voidaan asettaa niille sopiva vanhenemisaika ja välttää verkon ruuhkautumista vähemmän tärkeistä tai vanhentuneista viesteistä.

Testi 4: Resurssien käyttö

TMR-monilähetysjärjestelmän resurssienkulutus testattiin kuvan 4.10 mukaisella verkolla. Ensin testattiin muistin- ja prosessorinkäyttö lähetettäessä viestejä maksiminopeudella liikkuvalla solmulta 22. Viestejä lähetettiin kaikkiaan 1000 kappaletta.

Ensimmäisen testin tulokset on esitetty kuvassa 4.15. Prosessorinkäyttö nousee liikkuvalla solmulla ajoittain kymmeneen prosenttiin, mutta pysyy pääasiassa alle kuuden prosentin. Runkosolmulla prosessori kuormittuu odotetusti vähemmän, koska käytetty prosessori on tehokkaampi. Prosessorinkäyttö ei runkosolmulla ylitä 1,4 prosenttia.



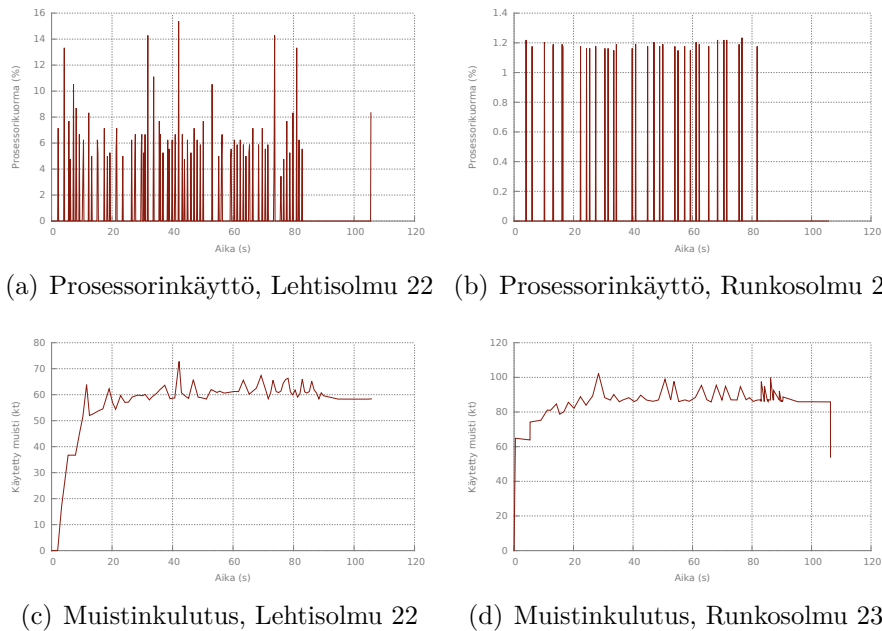
Kuva 4.15: Resurssienkulutus ensimmäisen aikana. Muistinkulutustestissä kului pitempi aika, koska muistinkulutusta tarkkaileva valgrind-ohjelma hidasti testin suoritusta.

Muistinkäyttö pysyy kohtuullisena molemmilla solmuilla. Alussa muistinkäyttö kasvaa hetken, koska kierronestoon käytetty Message-ID-kierronesto

tallettaa viestitunnisteita muistiin. Kun viestitunnisteiden maksimimäärä, 100 kappaletta, saavutetaan muistinkulutus pysyy suurin piirtein vakiona. Runkosolmun 23 muistinkäyttö on jonkun verran suurempaa kuin liikkuvan solmun, mutta kummallakaan solmulla muistia käytetään alle 120 kilotavua.

Toisessa testissä viestejä lähetetään viisi kappaletta yhdeltä solmulta kerrallaan niin, että solmujen välillä odotetaan yksi sekunti ja jokaiselta solmulta lähetetään 10 kertaa. Kaikkiaan lähetetään siis 800 viestiä. Testi simuloi tilannetta, jossa kaikki solmut verkossa lähettävät usein.

Testin tulokset on esitetty kuvaajassa 4.16. Sekä prosessorin-, että muistin- käyttö pysyvät kohtuullisena, kuten aiemmassakin testissä.



Kuva 4.16: Resurssienkulutus toisen testin aikana. Samoin kuin kuvaajan 4.15 testissä, muistinkulutustestissä kului pitempi aika, koska muistinkulutusta tarkkaileva valgrind-ohjelma hidasti testin suoritusta.

Testin tuloksena voidaan todeta, että TMR-monilähetysjärjestelmä kykenee toteuttamaan monilähetysten kohtuullisella resurssienkulutuksella. Resurssienkulutus on kohtuullista, vaikka käytetty laitteisto olisi suorituskyvyltään heikompaa. Resurssienkulutus on kohtuullista myös runkosolmulla, jotka ovat vastuussa viestien välittämisestä muille runkosolmuille.

Luku 5

Johtopäätökset

Tässä työssä on esitetty monilähetyksen teoriaa, yhden erityisvälitysverkoille tarkoitettun tiedonsiirtojärjestelmän toiminta, sekä monilähetysjärjestelmä erityisvälitysverkoille. Tehdyn työn perusteella voidaan antaa joitain suosituksia monilähetysjärjestelmän suunnittelusta ja toteutuksesta erityisvälitysverkoissa.

Työssä esitetty viestinvälitysjärjestelmä MICS on tätä työtä kirjoitettaessa vielä kehittyvä järjestelmä. Työssä esitetty monilähetysjärjestelmä on toimiva monilähetysjärjestelmä tämän hetken MICS-järjestelmään. Monilähetysjärjestelmää kannattaa kuitenkin kehittää, tai jopa toteuttaa täysin uusiksi, MICS-järjestelmän kehityessä.

MICS-järjestelmän dynaamisen reitityksen, sekä osoitetietojen jakamiseen liittyvien palvelujen kehittyessä, monilähetysjärjestelmä voidaan mahdollisesti toteuttaa yksinkertaisempuna, sekä tehokkaampana palveluna, joka myös kykenee palvelemaan verkon käyttäjiä paremmin.

Tässä työssä esitetyn TMR-monilähetysjärjestelmän toteutuksessa alla olevan verkon päälle rakennettiin erillinen monilähetysverkko monilähetyksiä varten. TMR-järjestelmä sisältää mekanismit viestinvälitykseen monilähetysverkossa, sekä käyttäjän poistumiseen ja siirtymiseen monilähetysverkon sisällä.

MICS-järjestelmän kehittyessä sen dynaamisen reitityksen ominaisuudet kehittyvät. TMR-järjestelmän kannattaa hyödyntää MICS-verkon dynaamista reititystä monilähetysviestien reitityksessä. Nykyisen toteutuksen erillisestä monilähetysverkosta voi olla edullista siirtyä malliin, jossa monilähetykset ovat osa MICS-järjestelmän reititystä.

Monilähetyksien ollessa osa MICS-reititystä, monilähetystä varten ei tarvita

erillistä MICS-verkon päälle toteutettua verkkoa. Monilähetystoiminnallisuuden lisääminen MICS-reititykseen voi olla haastava tehtävä, mutta kokonaisuuden kannalta verkon ylläpito muuttuu kevyemmäksi, koska MICS-verkon päällä ei tarvitse ylläpitää erillistä monilähetysverkkoa.

Toisaalta, kuten työssä esitettiin, TMR-monilähetysjärjestelmä kykenee tarjoamaan toimivan monilähetyspalvelun erityisvälitysverkoille. Järjestelmä ei myöskään käytä tuhlaavasti järjestelmän muisti-, sekä prosessori-resursseja.

Työn testeissä esitettiin TMR-järjestelmän toimivan myös liikkuvien reunasolmujen tapauksessa. Testausta voitaisiin laajentaa kattamaan suuren joukon liikkuvia reunasolmuja, sekä liikkuvia runkosolmuja, jotta voitaisiin tarkemmin arvioida järjestelmän kykyä reagoida verkon muutoksiin dynaamisesti.

Suuren osan verkosta muuttuessa jatkuvasti, monilähetysjärjestelmä hyötyisi matalan tason linkkitilätiedosta monilähetysreititystä suoritettaessa. Tämä puoltaisi MICS-järjestelmän sisään toteutettua monilähetyspalvelua. Toisaalta osittaista verkon tilaa voidaan välittää sovelluksena toteutetulle monilähetyspalvelulle esimerkiksi täsmälähetyksessä käytetyn reititystaulun kautta.

TMR-monilähetysjärjestelmä kykenee tarjoamaan toimivan monilähetysjärjestelmän erityisvälitysverkoille, mutta sitä kannattaa kehittää edelleen MICS-järjestelmän kehittyessä. Tarkemmat valinnat monilähetysjärjestelmän kehityksessä kannattaa tehdä, kun MICS-järjestelmän täsmäreititykseen käytetyt komponentit tukevat dynaamisia muutoksia kiitettävällä tasolla. Tällöin järjestelmän sisäiset komponentit ovat kehittyneet pidemmälle kohti lopullista muotoaan ja tiedetään paremmin täsmälähetyskomponenttien lopullinen tehtävä ja rajoitteet.

Viiteluettelo

- [1] Lxc, linux containers. Saatavissa: <http://linuxcontainers.org/>. Viitattu 13.1.2014.
- [2] ADAMS, A., NICHOLAS, J., AND SIADAK, W. Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised). RFC 3973, Internet Engineering Task Force, Jan. 2005.
- [3] BACCELLI, E., JACQUET, P., NGUYEN, D., AND CLAUSEN, T. OSPF Multipoint Relay (MPR) Extension for Ad Hoc Networks. RFC 5449, Internet Engineering Task Force, Feb. 2009.
- [4] CLARK, D. The design philosophy of the darpa internet protocols. *SIGCOMM Comput. Commun. Rev.* 18, 4 (Aug. 1988), 106–114.
- [5] CLAUSEN, T., DEARLOVE, C., AND DEAN, J. Mobile Ad Hoc Network (MANET) Neighborhood Discovery Protocol (NHDP). RFC 6130, Internet Engineering Task Force, Apr. 2011.
- [6] CLAUSEN, T., AND JACQUET, P. Optimized Link State Routing Protocol (OLSR). RFC 3626, Internet Engineering Task Force, Oct. 2003.
- [7] FALL, K. A delay-tolerant network architecture for challenged internets. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications* (New York, NY, USA, 2003), SIGCOMM '03, ACM, pp. 27–34.
- [8] FENNER, B., HANDLEY, M., HOLBROOK, H., AND KOUVELAS, I. Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised). RFC 4601, Internet Engineering Task Force, Aug. 2006. Updated by RFCs 5059, 5796, 6226.
- [9] FOUNDATION, P. S. Python ohjelmointikieli, 2014. Saatavissa: <http://python.org/>. Viitattu 13.1.2014.

- [10] FUHRMANN, T. T. Protocol independent multicast and asymmetric routing. Tech. rep., University of Mannheim, 2000.
- [11] HAAS, Z., AND PEARLMAN, M. The performance of query control schemes for the zone routing protocol. *Networking, IEEE/ACM Transactions on* 9, 4 (2001), 427–438.
- [12] JÄRVINEN, R., MÄÄTTÄ, J., LUOSTARINEN, R., LIUHTO, L., MANNER, J., TAIRA, T., AND LUOMA, M. A multi-interface dtn-like communication system. In *Wireless Conference 2011 - Sustainable Wireless Technologies (European Wireless), 11th European* (2011), pp. 1–8.
- [13] JÄRVINEN, R., MÄÄTTÄ, J., LUOSTARINEN, R., MANNER, J., AND LUOMA, M. Mics messaging platform: Architecture, design and routing. In *MILITARY COMMUNICATIONS CONFERENCE, 2010 - MILCOM 2010* (2010), pp. 1893–1898.
- [14] KO, Y.-B., AND VAIDYA, N. Geocasting in mobile ad hoc networks: location-based multicast algorithms. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on* (1999), pp. 101–110.
- [15] LAOUITI, A., JACQUET, P., MINET, P., VIENNOT, L., CLAUSEN, T., AND ADJIH, C. Multicast Optimized Link State Routing. Research Report RR-4721, INRIA, 2003.
- [16] MACKER, J. Simplified Multicast Forwarding. RFC 6621, Internet Engineering Task Force, May 2012.
- [17] MACKER, J., DEAN, J., AND CHAO, W. Simplified multicast forwarding in mobile ad hoc networks. In *Military Communications Conference, 2004. MILCOM 2004. 2004 IEEE* (2004), vol. 2, pp. 744–750 Vol. 2.
- [18] MALKIN, G. RIP Version 2. RFC 2453, Internet Engineering Task Force, Nov. 1998. Updated by RFC 4822.
- [19] MERAIHI NAIMI, A., ADJIH, C., MINET, P., AND RODOLAKIS, G. Simulation-based comparison of three wireless multicast routing protocols: Most, molsr and smolr. In *Ad-Hoc, Mobile and Wireless Networks*, I. Nikolaidis and K. Wu, Eds., vol. 6288 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2010, pp. 1–14.

- [20] MOY, J. OSPF Version 2. RFC 2328, Internet Engineering Task Force, Apr. 1998. Updated by RFCs 5709, 6549, 6845, 6860.
- [21] NS 3 PROJECT. ns-3 network simulator, 2014. Saatavissa: <http://nsnam.org/>.
- [22] PENNINGTON, H., CARLSSON, A., AND LARSSON, A. D-bus specification 0.12, 2006. Saatavissa: <http://dbus.freedesktop.org/doc/dbus-specification.html>.
- [23] PERKINS, C., BELDING-ROYER, E., AND DAS, S. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561, Internet Engineering Task Force, July 2003.
- [24] PERKINS, C. E., AND BHAGWAT, P. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In *Proceedings of the conference on Communications architectures, protocols and applications* (New York, NY, USA, 1994), SIGCOMM '94, ACM, pp. 234–244.
- [25] REKHTER, Y., LI, T., AND HARES, S. A Border Gateway Protocol 4 (BGP-4). RFC 4271, Internet Engineering Task Force, Jan. 2006. Updated by RFCs 6286, 6608, 6793.
- [26] RESNICK, P. Internet Message Format. RFC 5322, Internet Engineering Task Force, Oct. 2008.
- [27] RITVANEN, K. Multicast routing and addressing. Tech. rep., Department of Computer Science and Engineering, Helsinki University of Technology, 2004.
- [28] RODOLAKIS, G., NAIMI, A. M., AND LAOUITI, A. Multicast overlay spanning tree protocol for ad hoc networks. In *Proceedings of the 5th international conference on Wired/Wireless Internet Communications* (Berlin, Heidelberg, 2007), WWIC '07, Springer-Verlag, pp. 290–301.
- [29] ROYER, E. M., AND PERKINS, C. E. Multicast operation of the ad-hoc on-demand distance vector routing protocol. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking* (New York, NY, USA, 1999), MobiCom '99, ACM, pp. 207–218.
- [30] RÜHRUP, S. Theory and practice of geographic routing. In *Ad Hoc and Sensor Wireless Networks: Architectures, Algorithms and Protocols*, Y.-W. L. Hai Liu, Xiaowen Chu, Ed. Bentham Science, 2009.

- [31] SALIM, J., KHOSRAVI, H., KLEEN, A., AND KUZNETSOV, A. Linux Netlink as an IP Services Protocol. RFC 3549, Internet Engineering Task Force, July 2003.
- [32] VAHDAT, A., AND BECKER, D. Epidemic routing for partially-connected ad hoc networks. Tech. rep., Department of Computer Science, Duke University, 2000.
- [33] YAO, P., KROHNE, E., AND CAMP, T. Performance comparison of geocast routing protocols for a manet. In *Computer Communications and Networks, 2004. ICCCN 2004. Proceedings. 13th International Conference on* (2004), pp. 213–220.